Effective Error Channels for Magic State Distillation

Aiden Karpf

2025 Global Quantum Leap Internship

Long Update

Outline

I. Introduction

- Research question / motivation
- Magic state distillation
- Steane encoding

Outline

I. Introduction
- Research question / motivation
- Magic state distillation
- Steane encoding

II. Effective Error Channels and Results
- Process tomography
- Simulating logical circuit with effective error channels

Outline

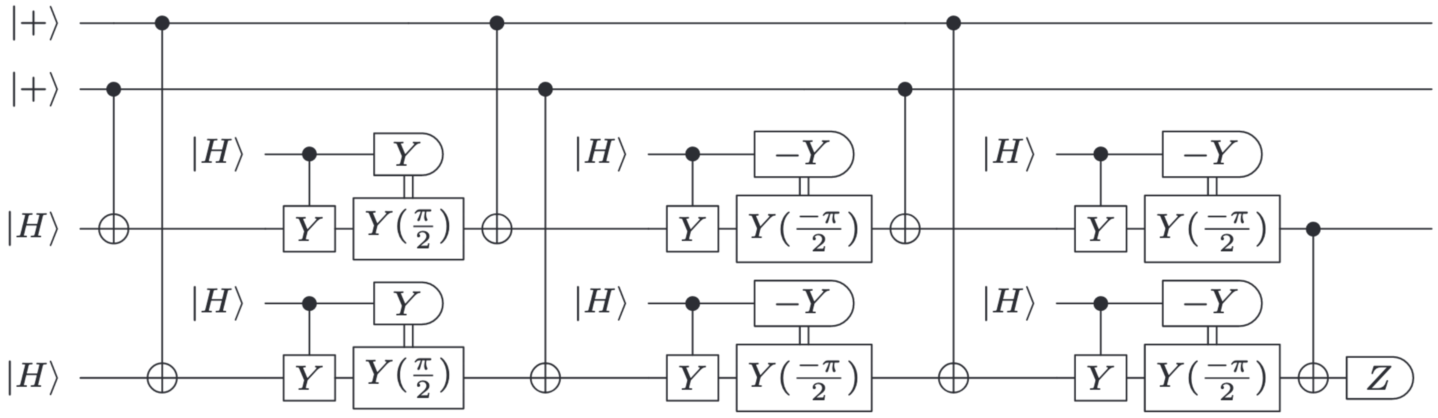
I. Introduction
- Research question / motivation
- Magic state distillation
- Steane encoding

II. Effective Error Channels and Results
- Process tomography
- Simulating logical circuit with effective error channels

III. Outlook
- Simulating encoded circuit with base code for comparison?

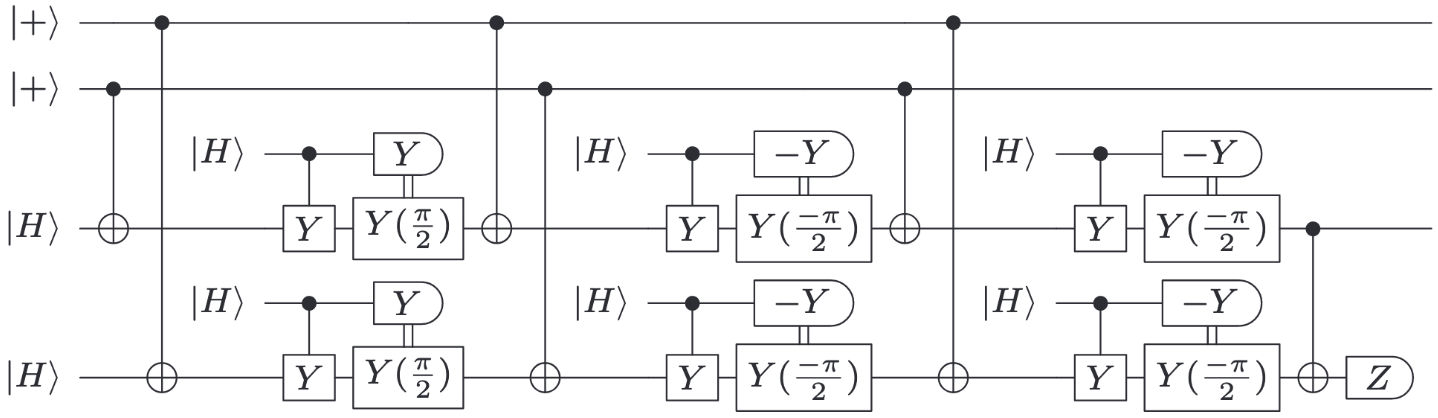
I. Introduction

Research Question

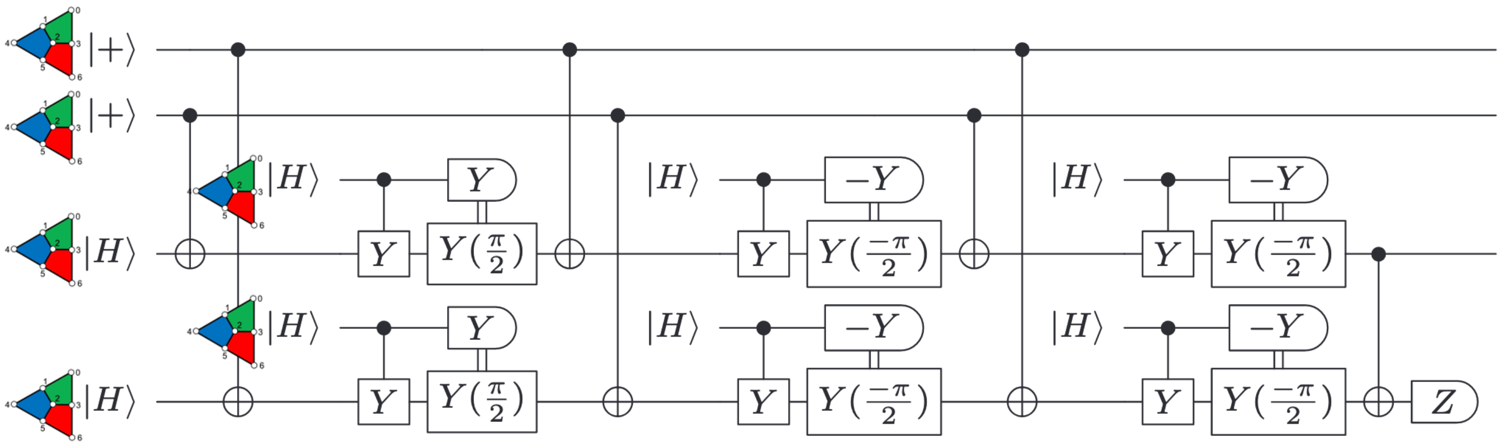


Logical Circuit (6 qubits)

Research Question

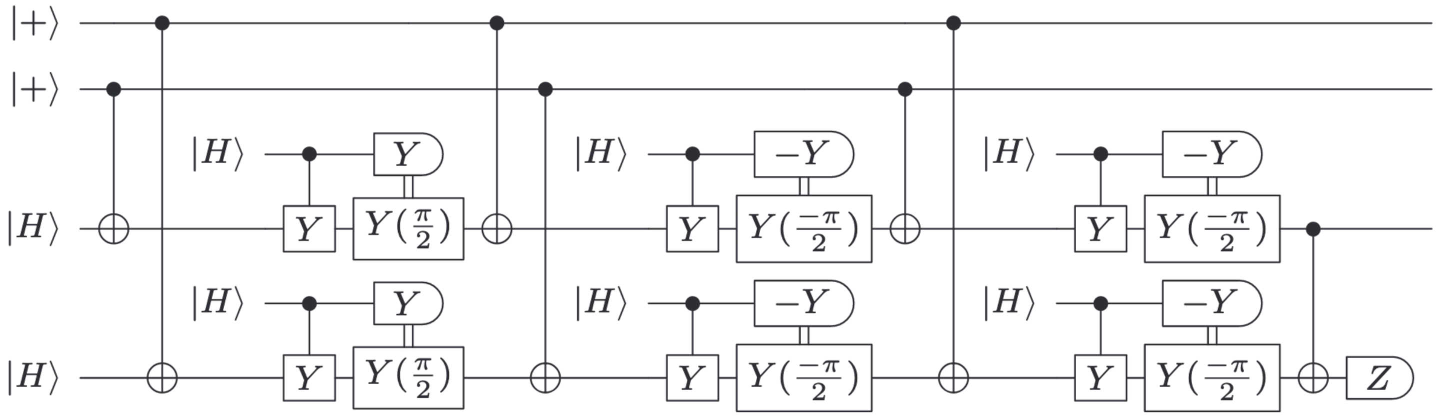


Logical Circuit (6 qubits)

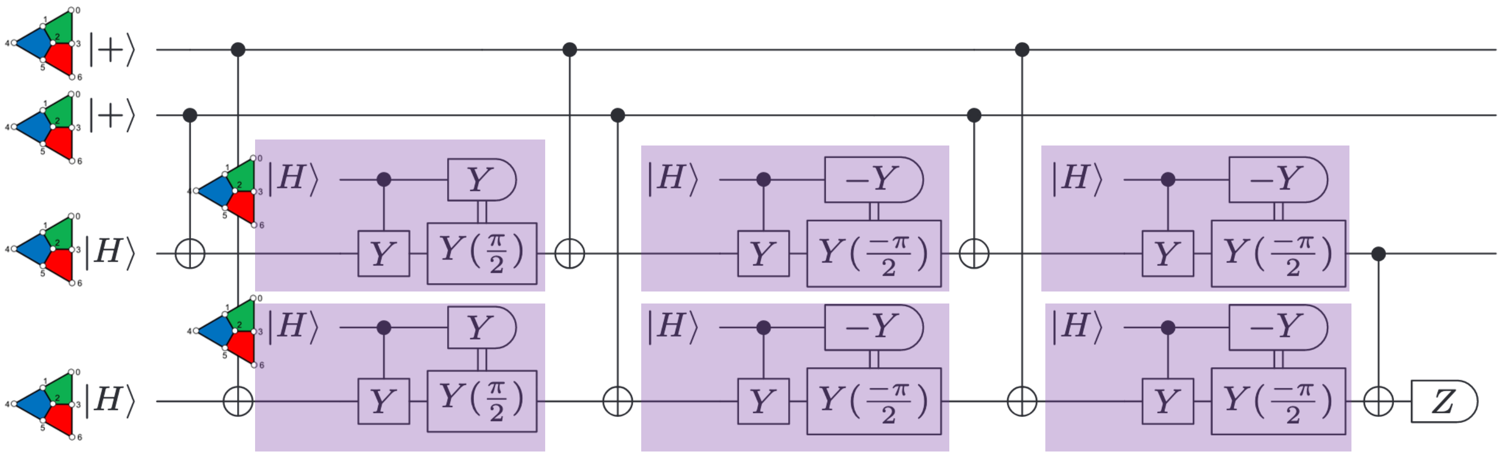


Encoded circuit with Steane code (50 qubits)

Research Question



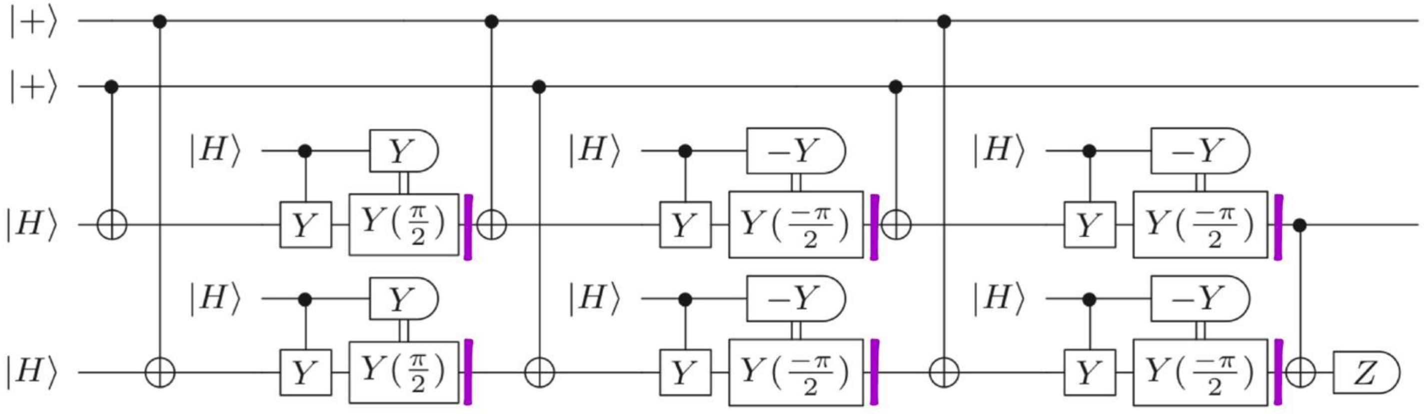
Logical Circuit (6 qubits)



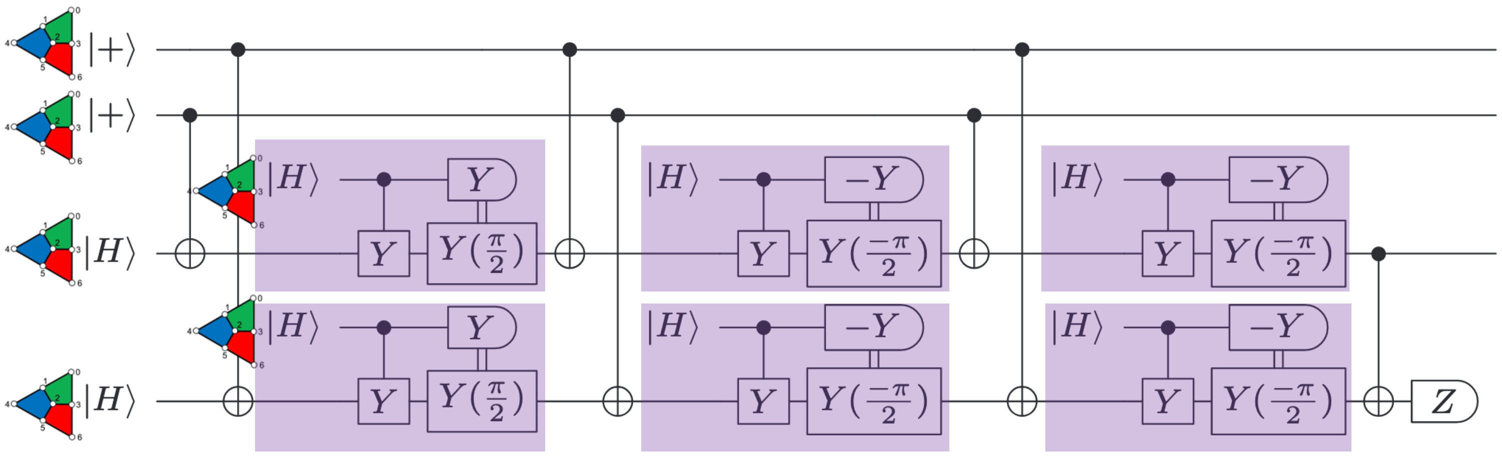
Encoded circuit with Steane code (50 qubits)

Noise depends on Steane-encoding !

Research Question



Logical Circuit (6 qubits)

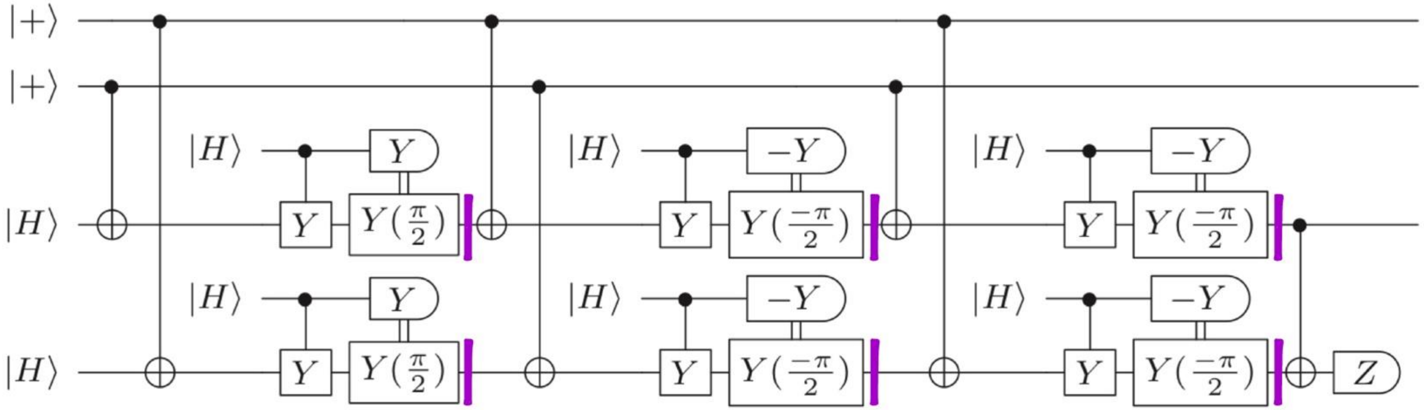


Encoded circuit with Steane code (50 qubits)

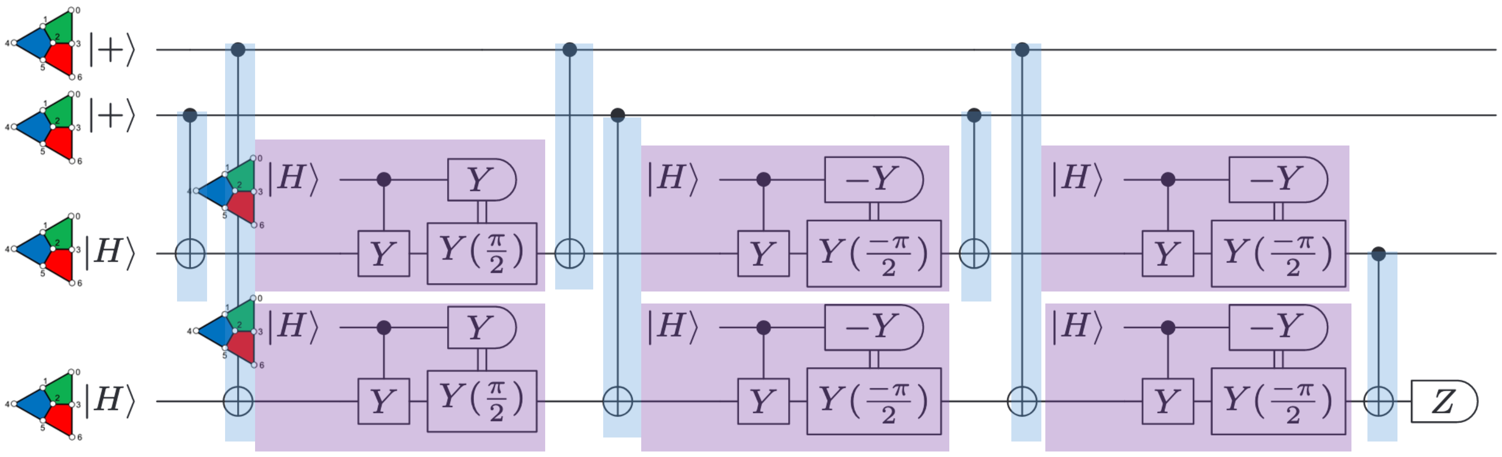
Noise depends on Steane-encoding !

Research Question

= effective error channel 1



Logical Circuit (6 qubits)



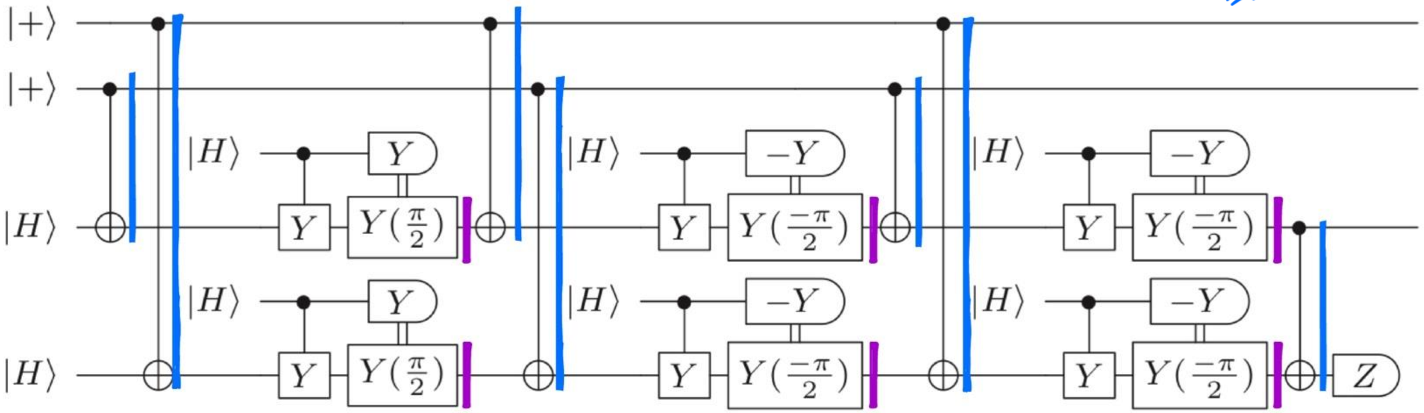
Encoded circuit with Steane code (50 qubits)

Noise depends on Steane-encoding !

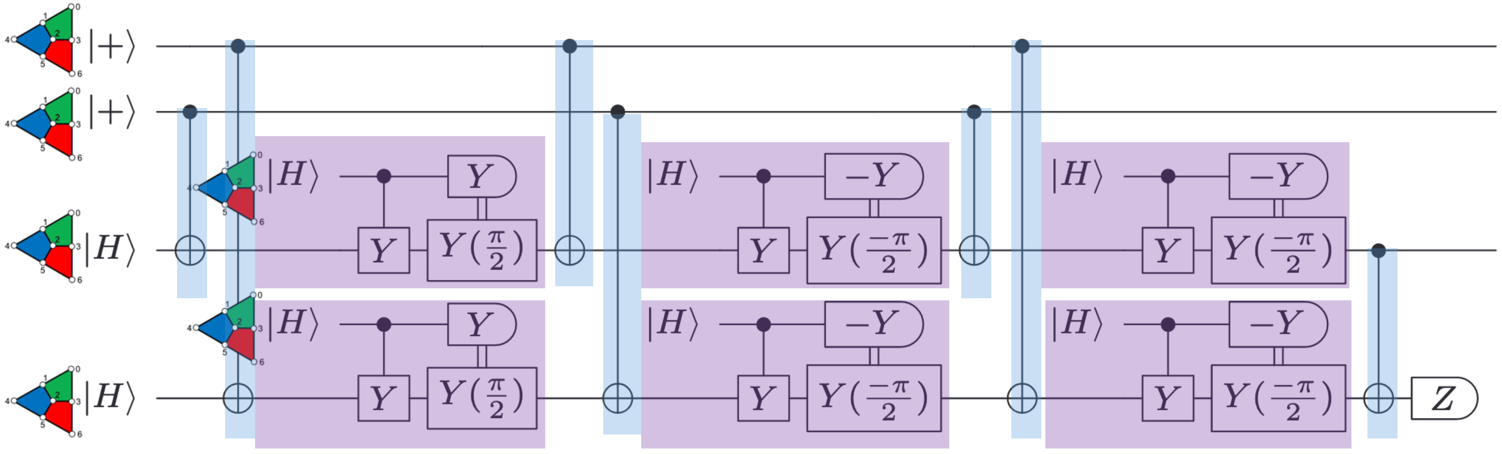
Research Question

= effective error channel 1

= effective error channel 2



Logical Circuit (6 qubits)



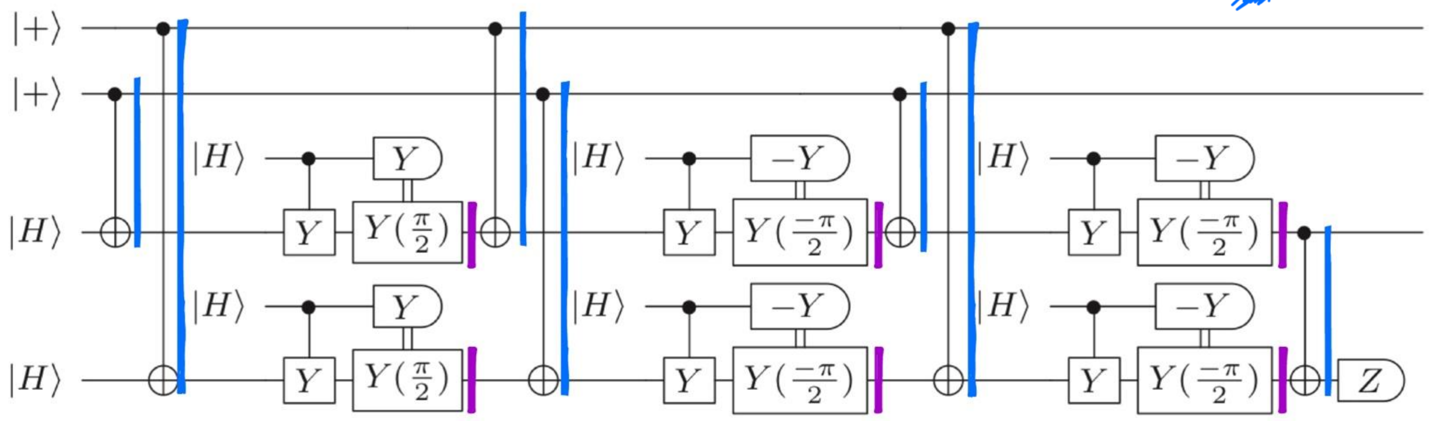
Encoded circuit with Steane code (50 qubits)

Noise depends on Steane-encoding !

Research Question

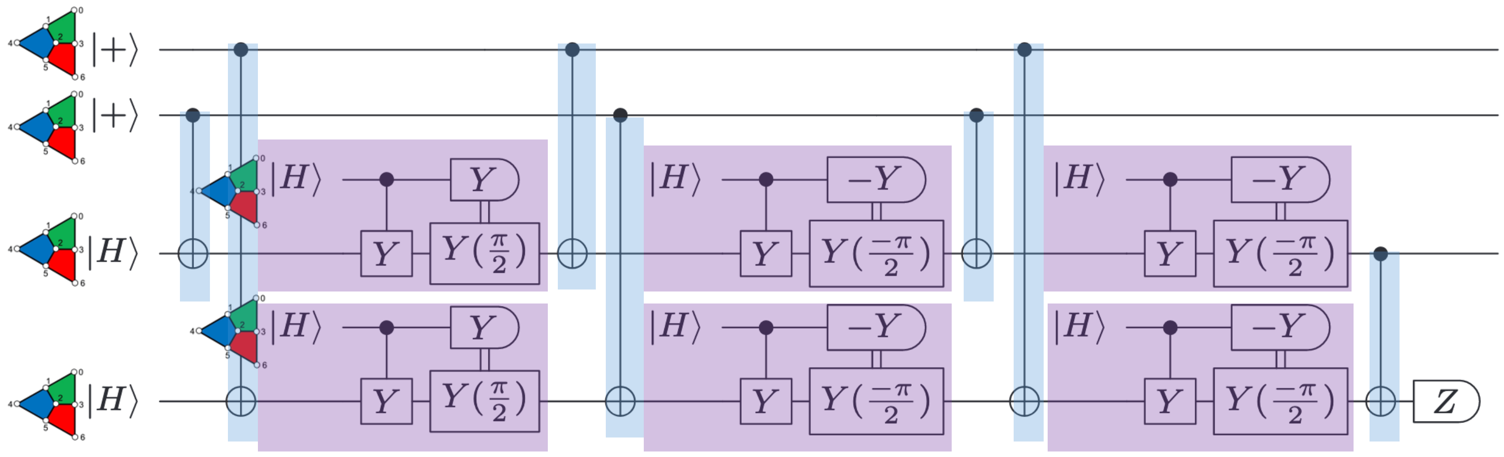
= effective error channel 1

= effective error channel 2



Logical Circuit (6 qubits)

Logical circuit with effective error channels

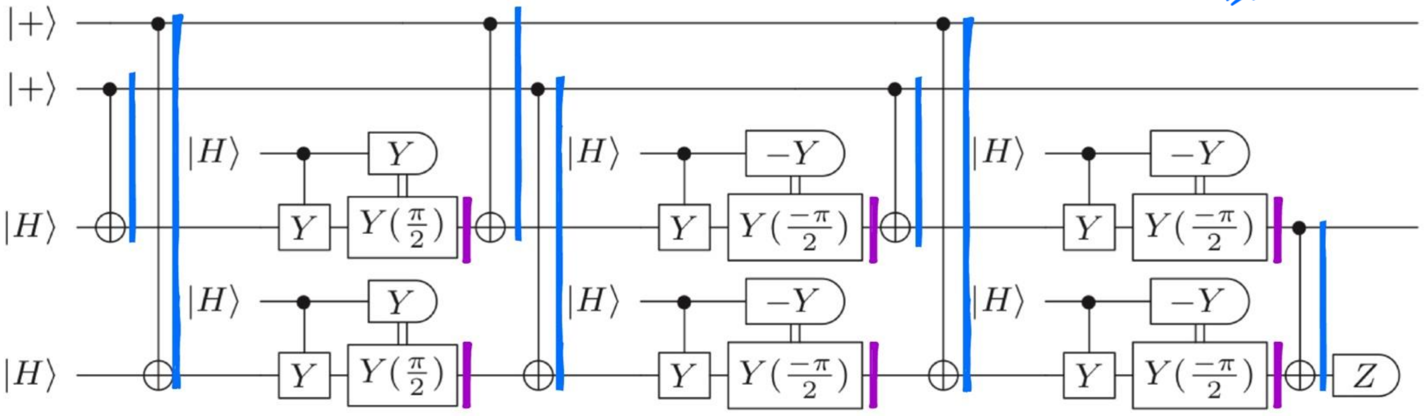


Encoded circuit with Steane code (50 qubits)

Research Question

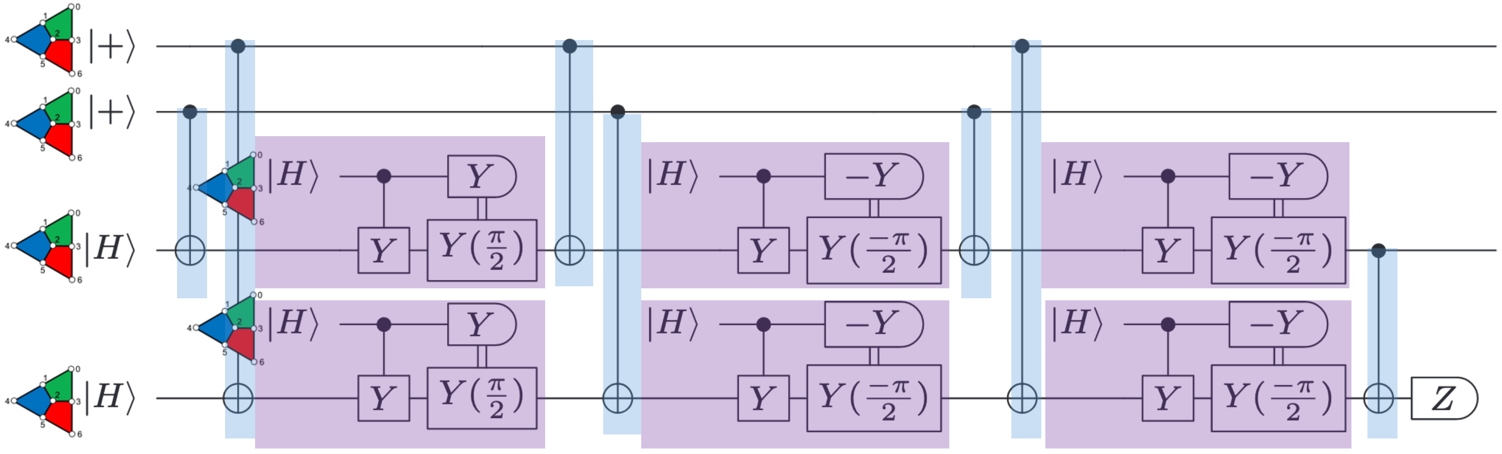
= effective error channel 1

= effective error channel 2



Logical Circuit (6 qubits)

Logical circuit with effective error channels



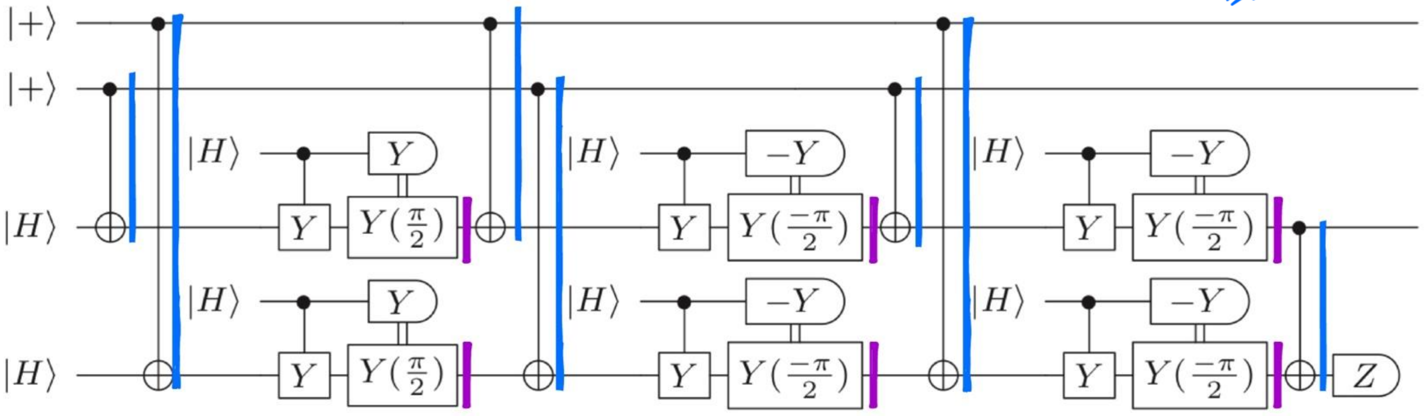
Encoded circuit with Steane code (50 qubits)

Fully encoded circuit with base code

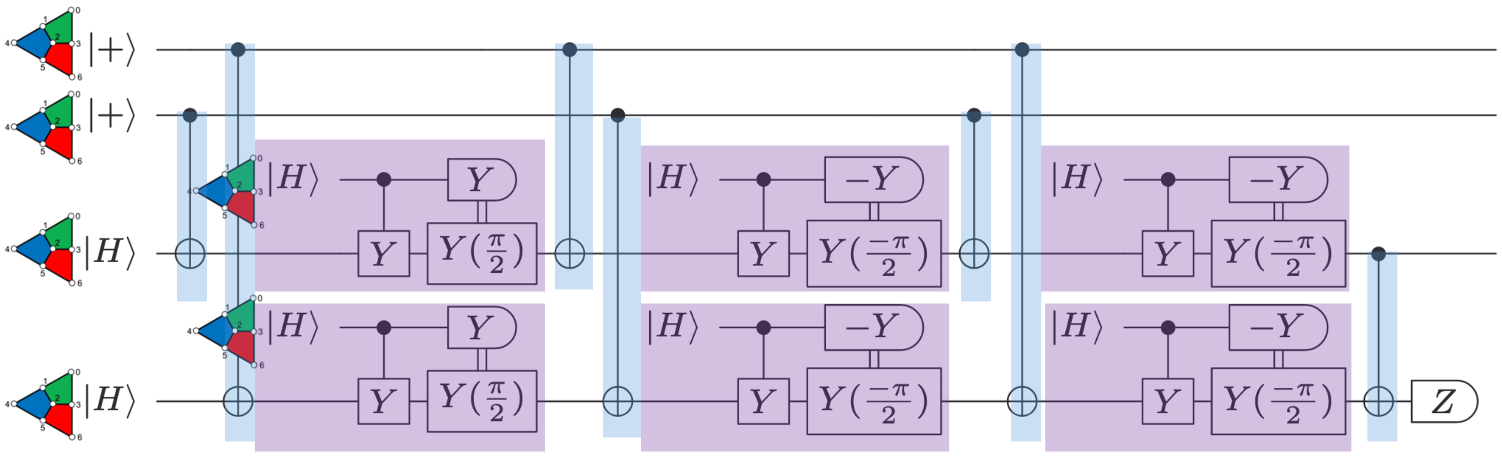
Research Question


= effective error channel 1

= effective error channel 2



Logical Circuit (6 qubits)



Encoded circuit with Steane code (50 qubits)

Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Fully encoded circuit with base code

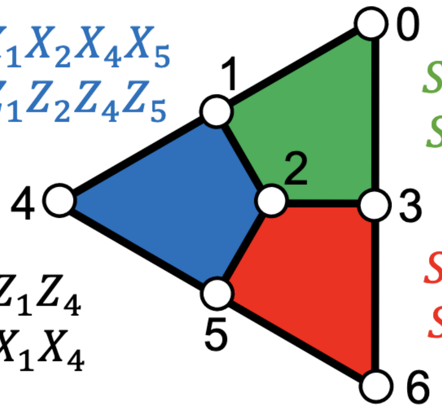
Why QEC?

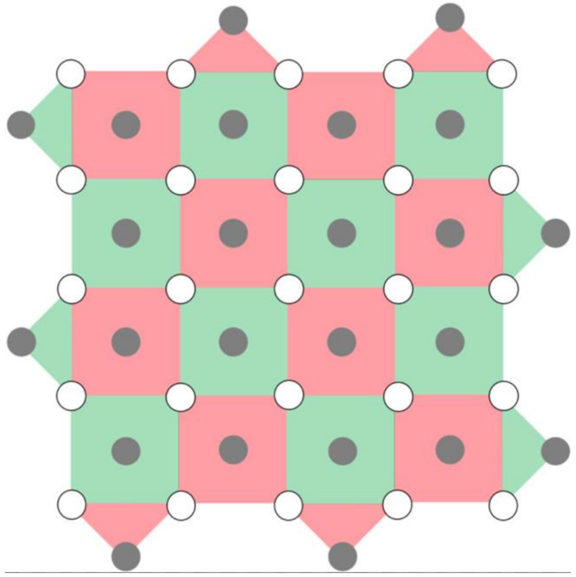
$S_{X1} = X_1X_2X_4X_5$
$S_{Z1} = Z_1Z_2Z_4Z_5$

$S_{X0} = X_0X_1X_2X_3$
$S_{Z0} = Z_0Z_1Z_2Z_3$

$S_{X2} = X_2X_3X_5X_6$
$S_{Z2} = Z_2Z_3Z_5Z_6$

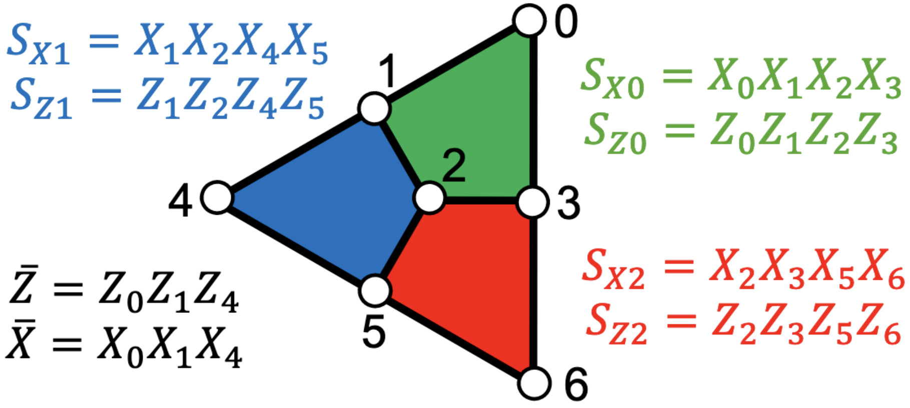
$\bar{Z} = Z_0Z_1Z_4$
$\bar{X} = X_0X_1X_4$

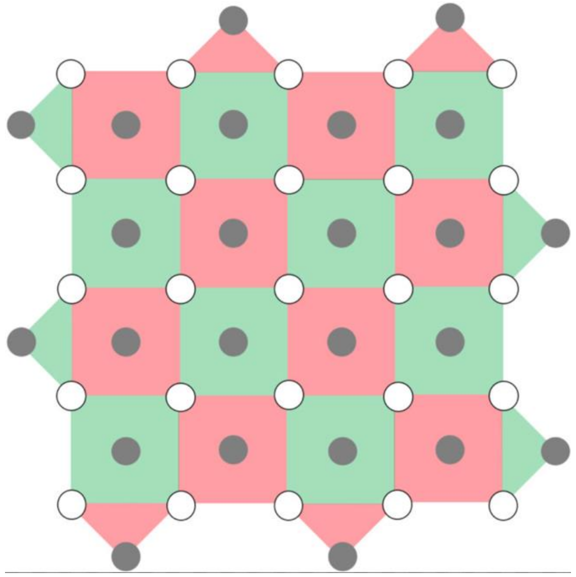




Why QEC?

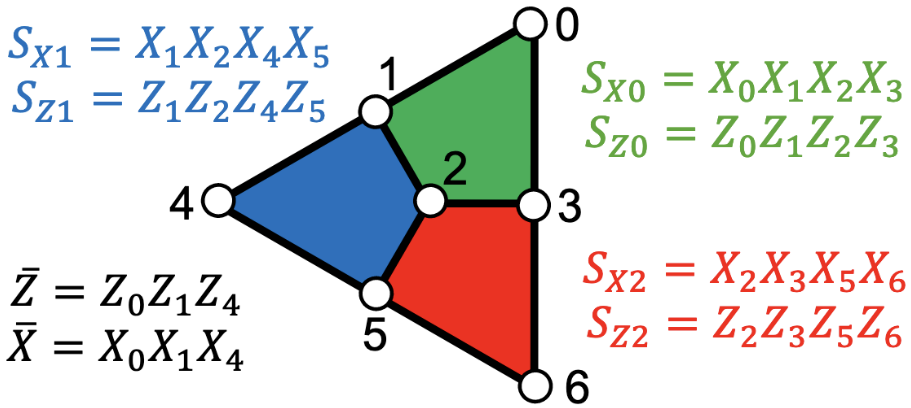
- Physical error rates in experiment: ~10^{-4}

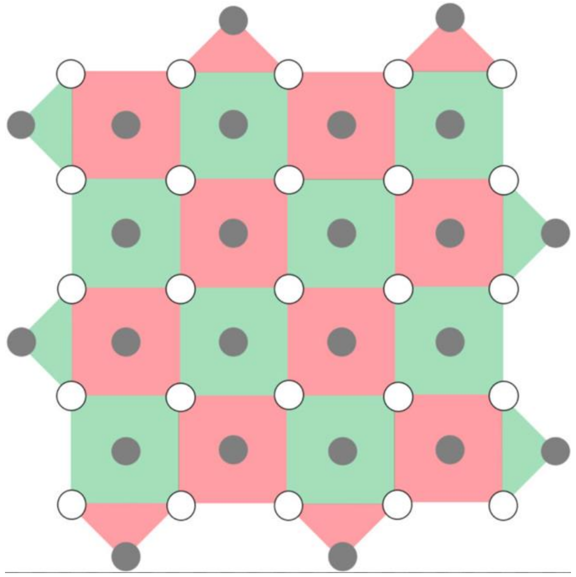




Why QEC?

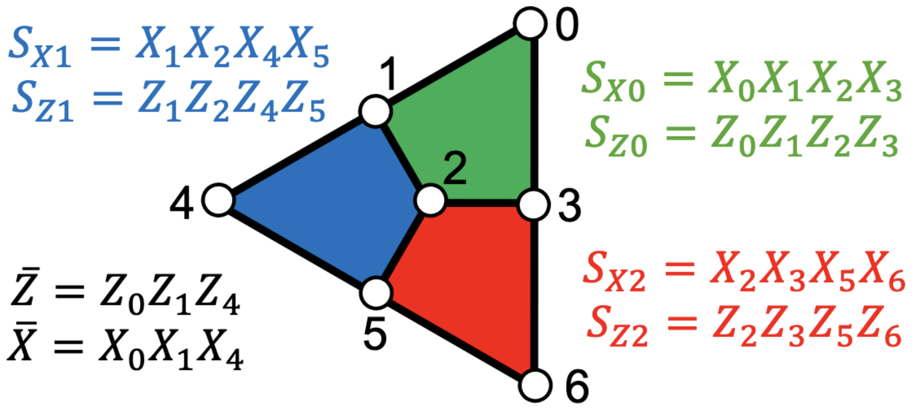
- Physical error rates in experiment: ~10^{-4}
- Logical error rates required for algorithms: ~10^{-15}

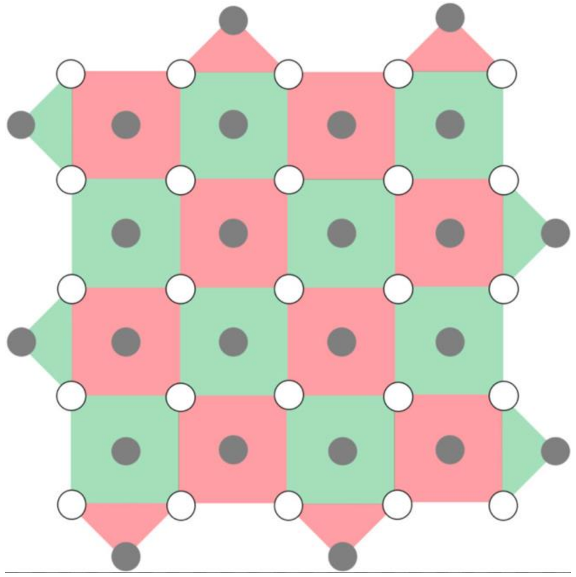




Why QEC?

- Physical error rates in experiment: $\sim 10^{-4}$
- Logical error rates required for algorithms: $\sim 10^{-15}$
- QEC allows us to close the gap





Why Magic States?

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set
- No QEC code can transversally implement a universal gate set *(Eastin-Knill)*

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set
- No QEC code can transversally implement a universal gate set *(Eastin-Knill)*
- Magic states can be injected to realize a non-Clifford gate

Why Magic States?

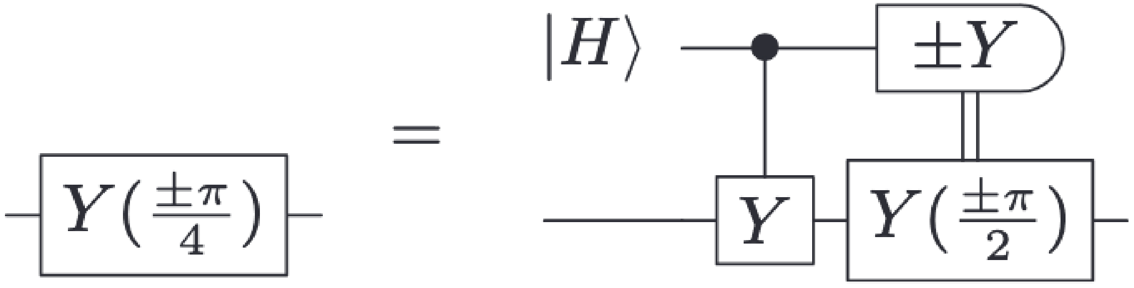
- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set
- No QEC code can transversally implement a universal gate set *(Eastin-Knill)*
- Magic states can be injected to realize a non-Clifford gate

$$|H\rangle = Y\left(\tfrac{\pi}{4}\right)|0\rangle = \cos\left(\frac{\pi}{8}\right)|0\rangle + \sin\left(\frac{\pi}{8}\right)|1\rangle$$

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set
- No QEC code can transversally implement a universal gate set *(Eastin-Knill)*
- Magic states can be injected to realize a non-Clifford gate

$$|H\rangle = Y\left(\tfrac{\pi}{4}\right)|0\rangle = \cos\left(\frac{\pi}{8}\right)|0\rangle + \sin\left(\frac{\pi}{8}\right)|1\rangle$$

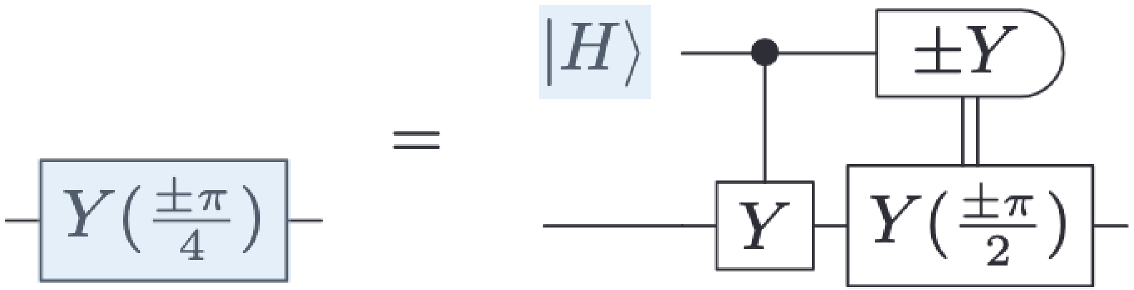


Magic state injection

Why Magic States?

- Clifford circuits can be simulated efficiently *(Gottesman-Knill)*
- At least one non-Clifford gate is required for a universal gate set
- No QEC code can transversally implement a universal gate set *(Eastin-Knill)*
- Magic states can be injected to realize a non-Clifford gate

$$|H\rangle = Y\left(\tfrac{\pi}{4}\right)|0\rangle = \cos\left(\tfrac{\pi}{8}\right)|0\rangle + \sin\left(\tfrac{\pi}{8}\right)|1\rangle$$



Magic state injection

Why Distillation?

Why Distillation?

- Physical preparation of magic states (e.g., $|H\rangle$) are noisy

Why Distillation?

- Physical preparation of magic states (e.g., $|H\rangle$) are noisy
- Distillation allows us to obtain higher fidelity states

Why Distillation?

- Physical preparation of magic states (e.g., $|H\rangle$) are noisy
- Distillation allows us to obtain higher fidelity states

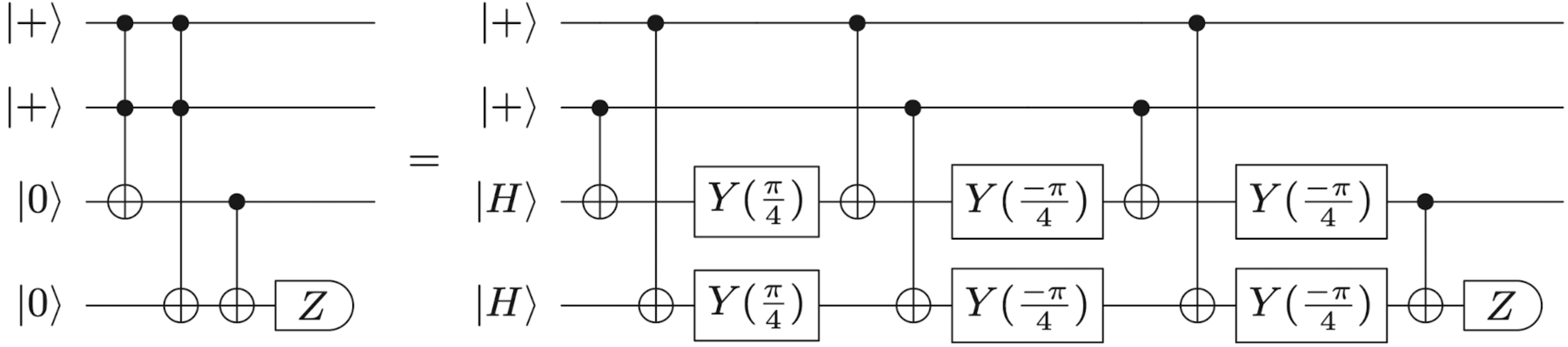


FIG. 1. $|H\rangle$- to- $|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013)*

Why Distillation?

- Physical preparation of magic states (e.g., $|H\rangle$) are noisy
- Distillation allows us to obtain higher fidelity states

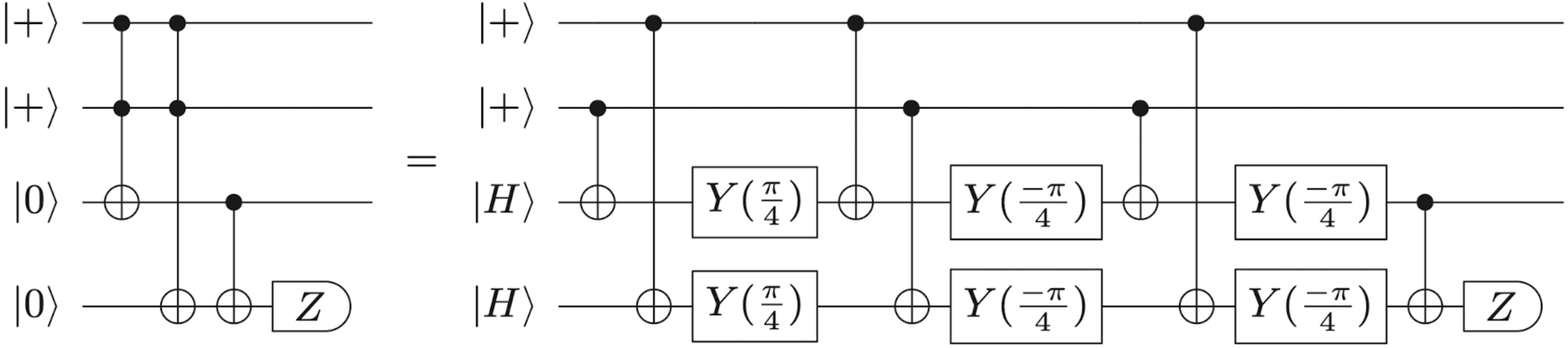


FIG. 1. $|H\rangle$ - to- $|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013)*

What's going on here?

Relevant Decompositions and Identities

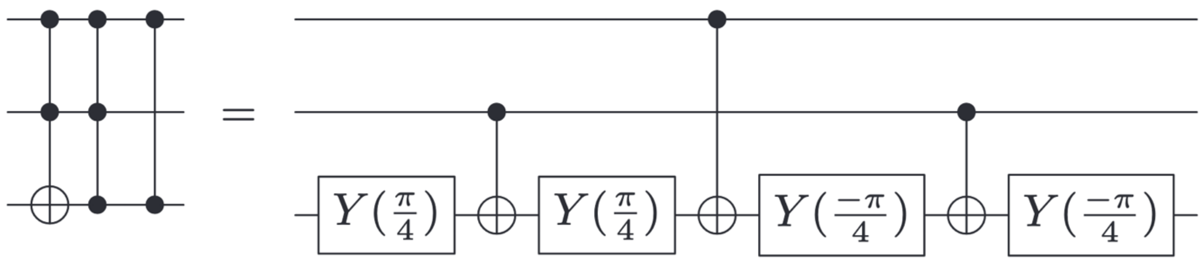


FIG. 2

Relevant Decompositions and Identities

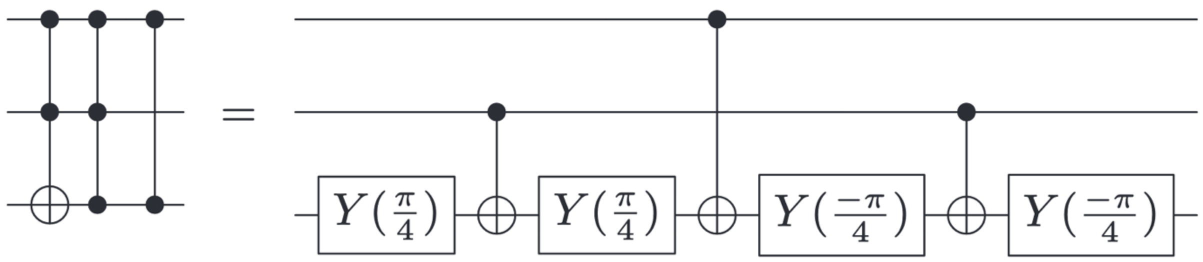


FIG. 2

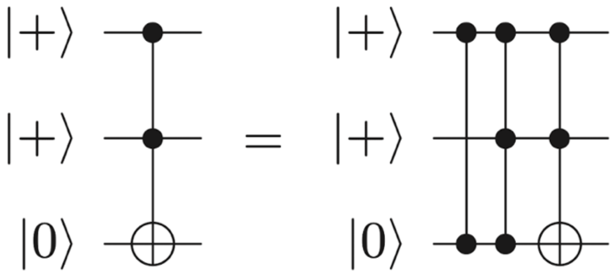


FIG. 3

Relevant Decompositions and Identities

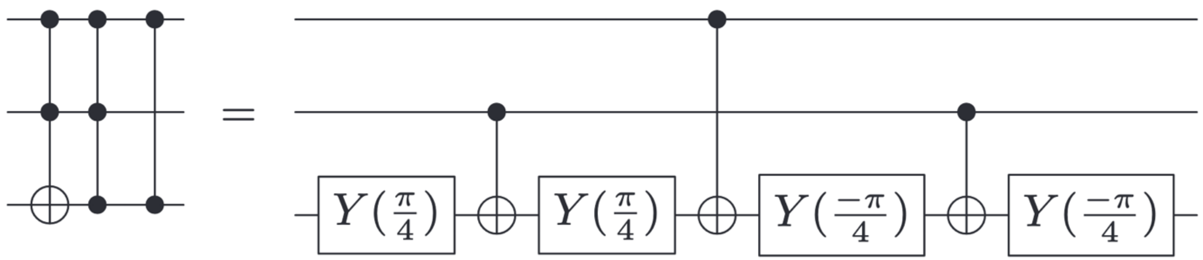


FIG. 2

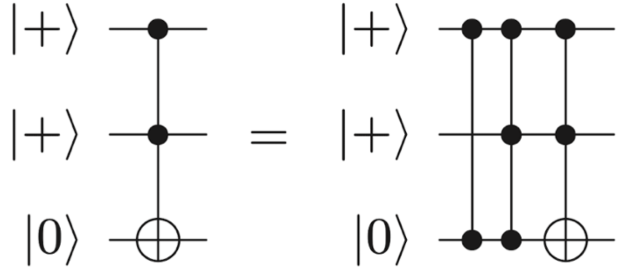


FIG. 3

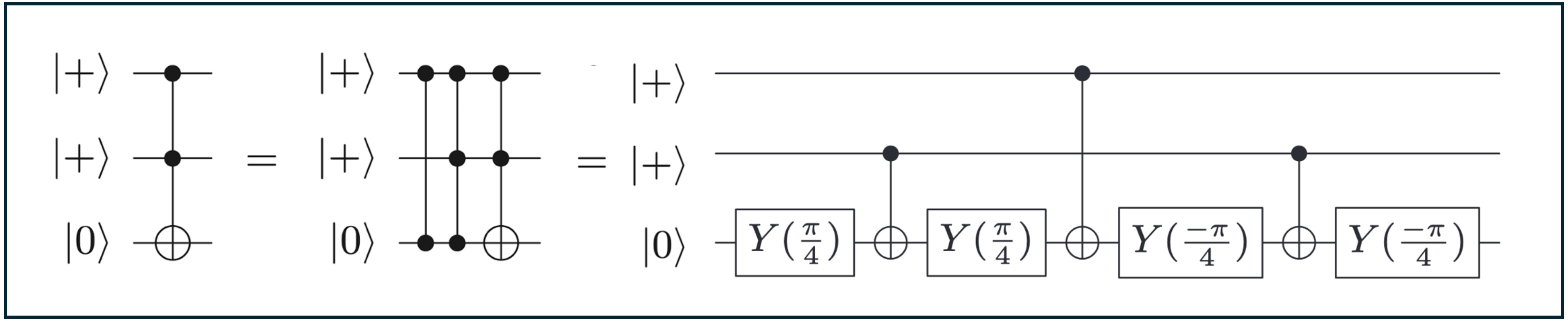


FIG. 2 + 3

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

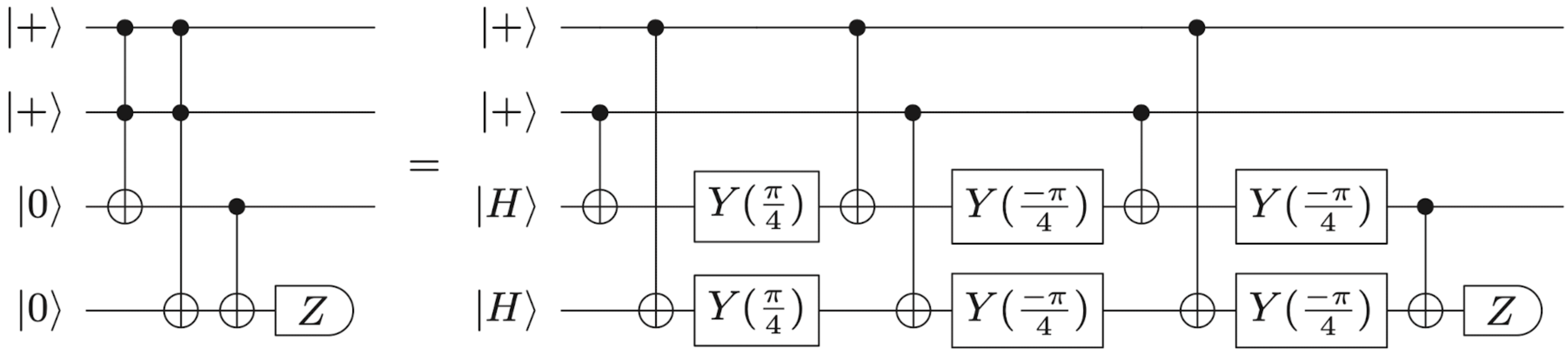


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

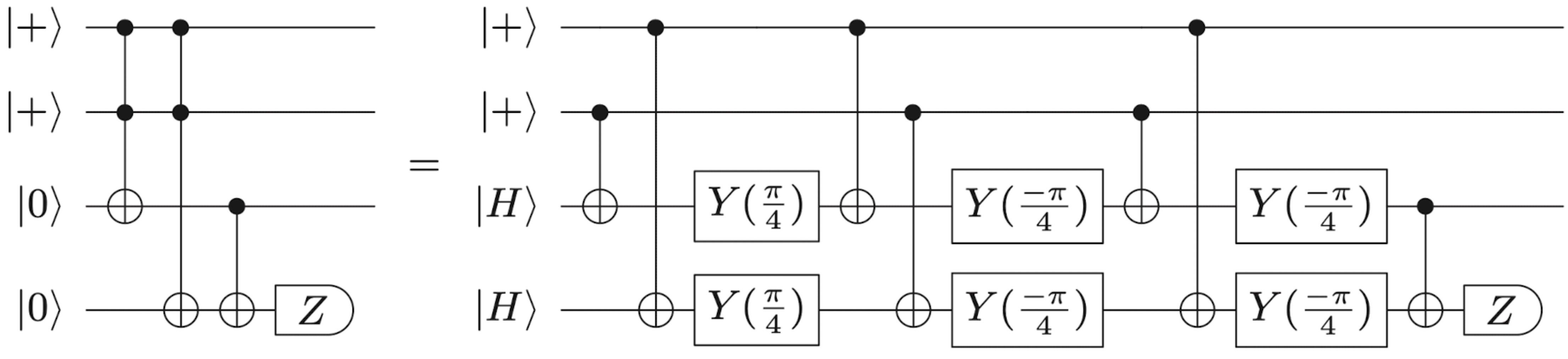


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

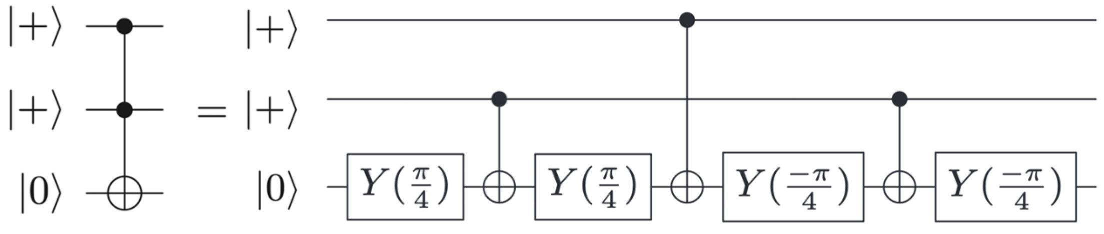


FIG. 2 + 3

$|H\rangle$-to-$|\text{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013)*

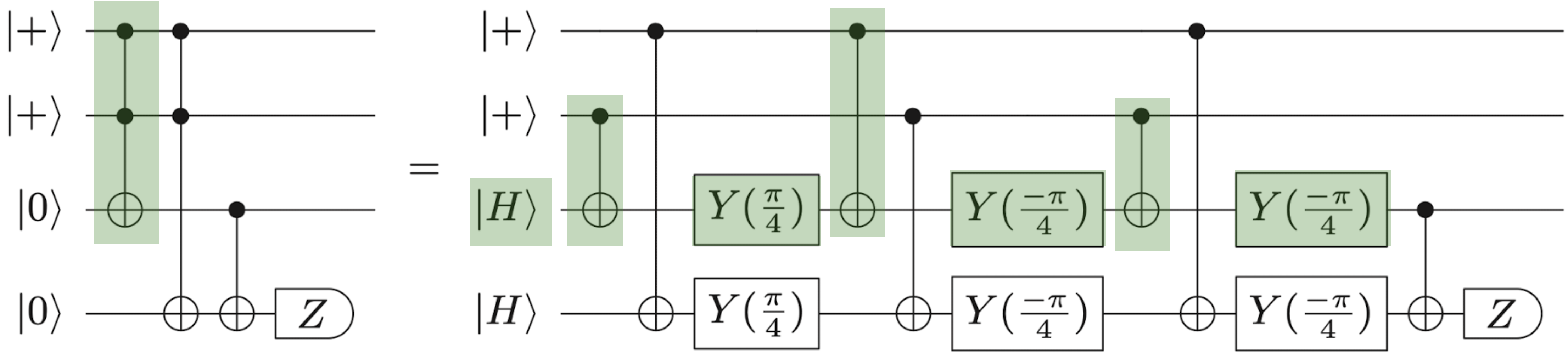


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

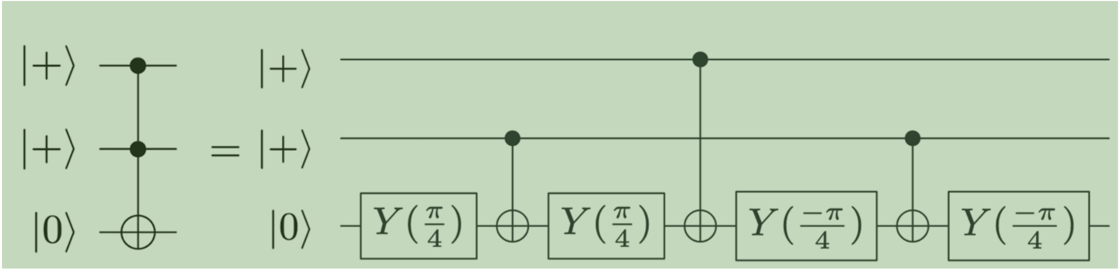


FIG. 2 + 3

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

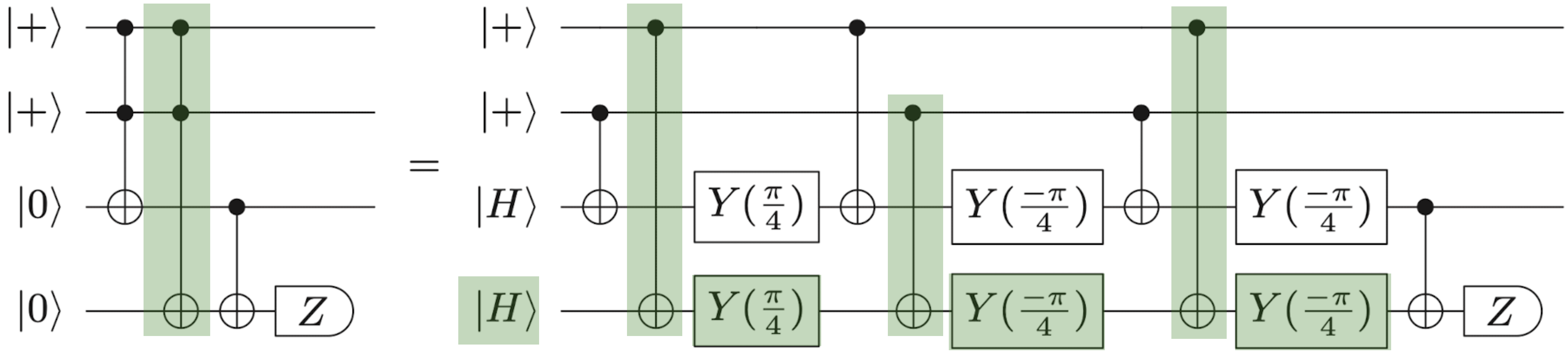


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

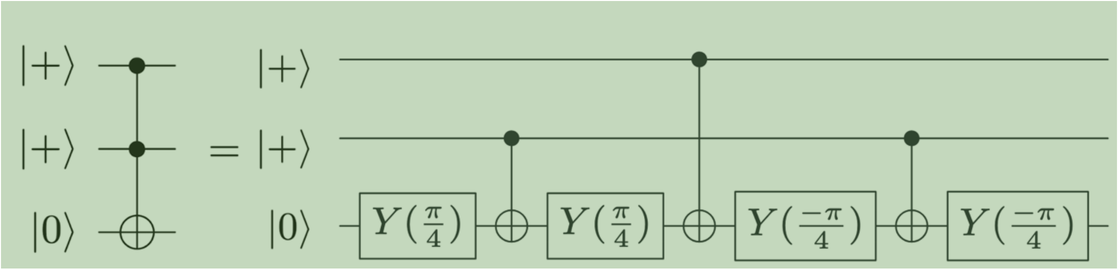


FIG. 2 + 3

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

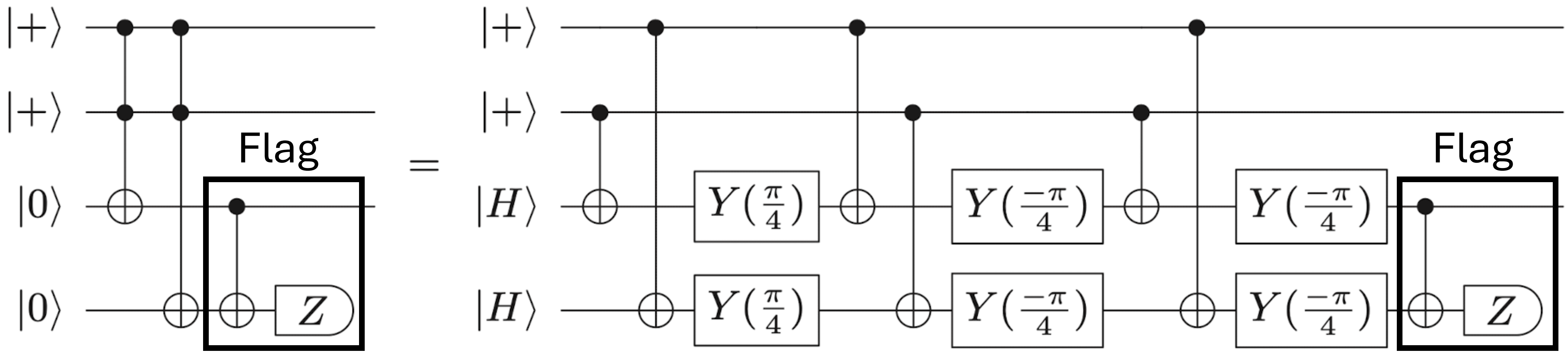


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

$|H\rangle$-to-$|\text{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

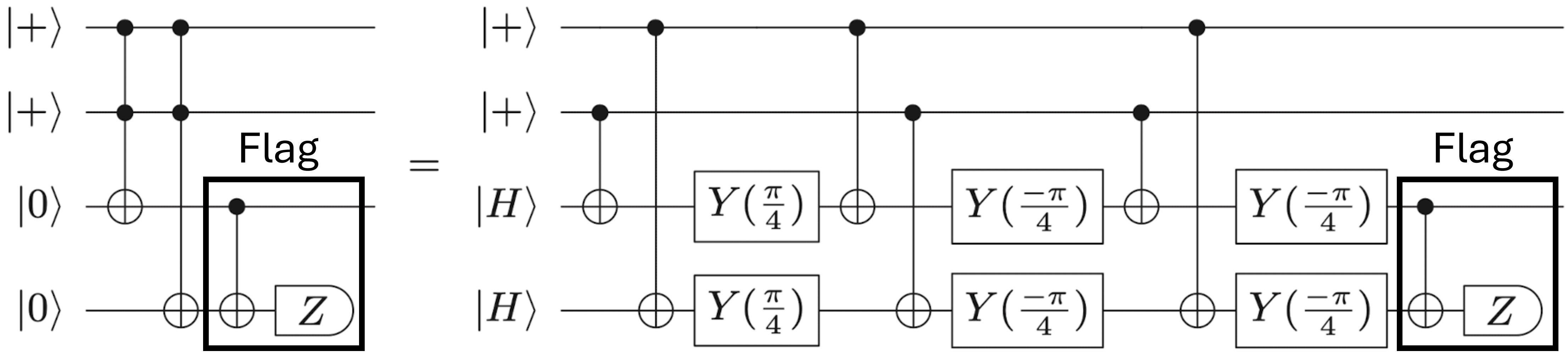


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

Output acceptance rate of the circuit is $a(p) = 1 - 8p + \mathcal{O}(p^2)$

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

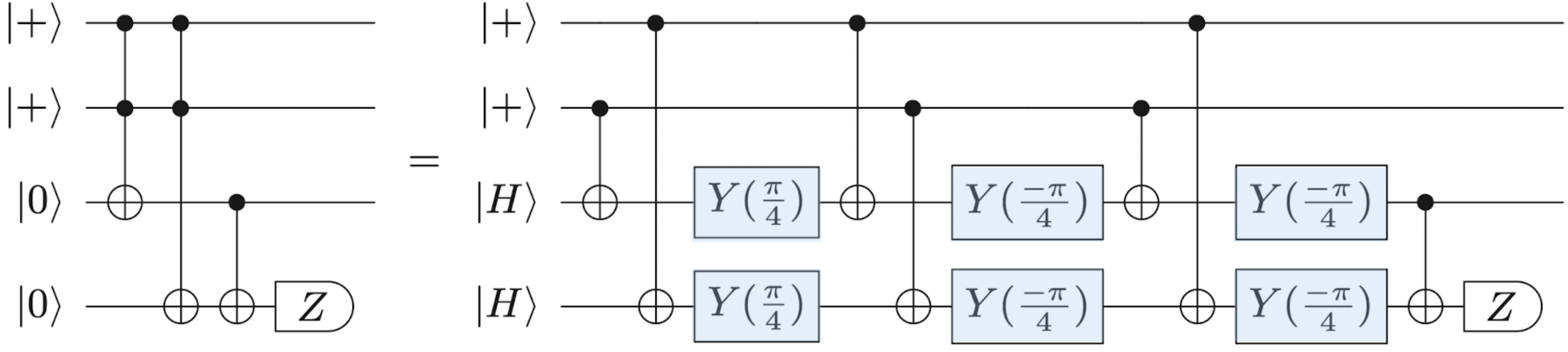


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

But how do we realize these non-Clifford $Y(\pm\pi/4)$ rotations?

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013)*

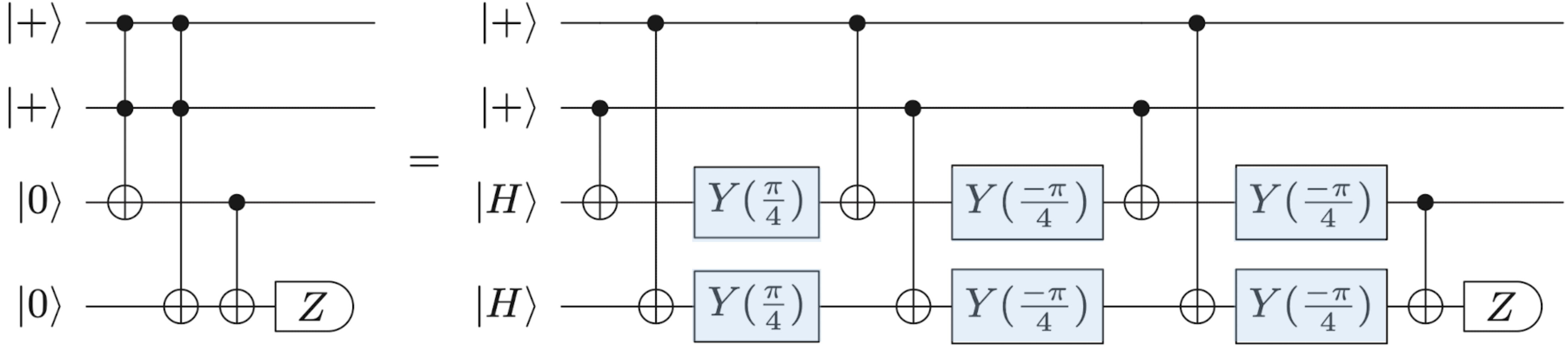


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

But how do we realize these non-Clifford $Y(\pm\pi/4)$ rotations?

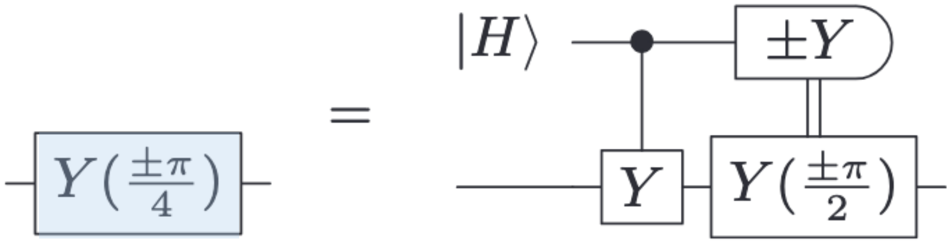


FIG. 4

$|H\rangle$-to-$|\mathrm{Toffoli}\rangle$ Distillation Circuit (*Eastin 2013*)

= *non-Clifford*

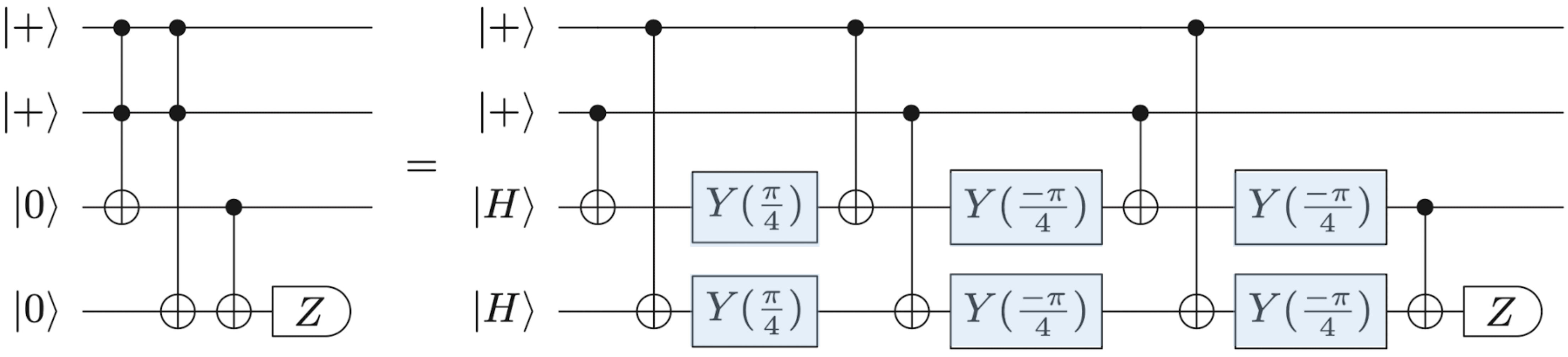


FIG. 1. *H*-to-Toffoli Distillation circuit. The output is discarded whenever a non-trivial outcome is obtained.

But how do we realize these non-Clifford $Y(\pm\pi/4)$ rotations?

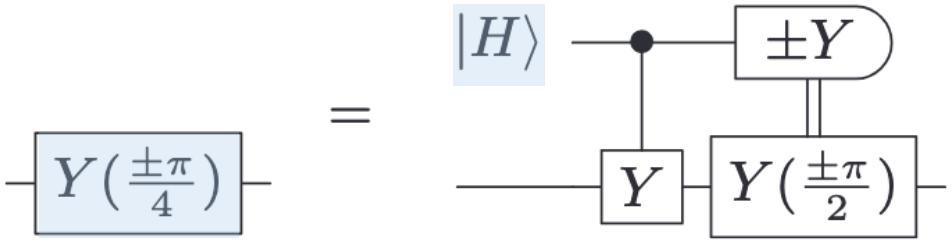


FIG. 4

Full Logical Distillation Circuit

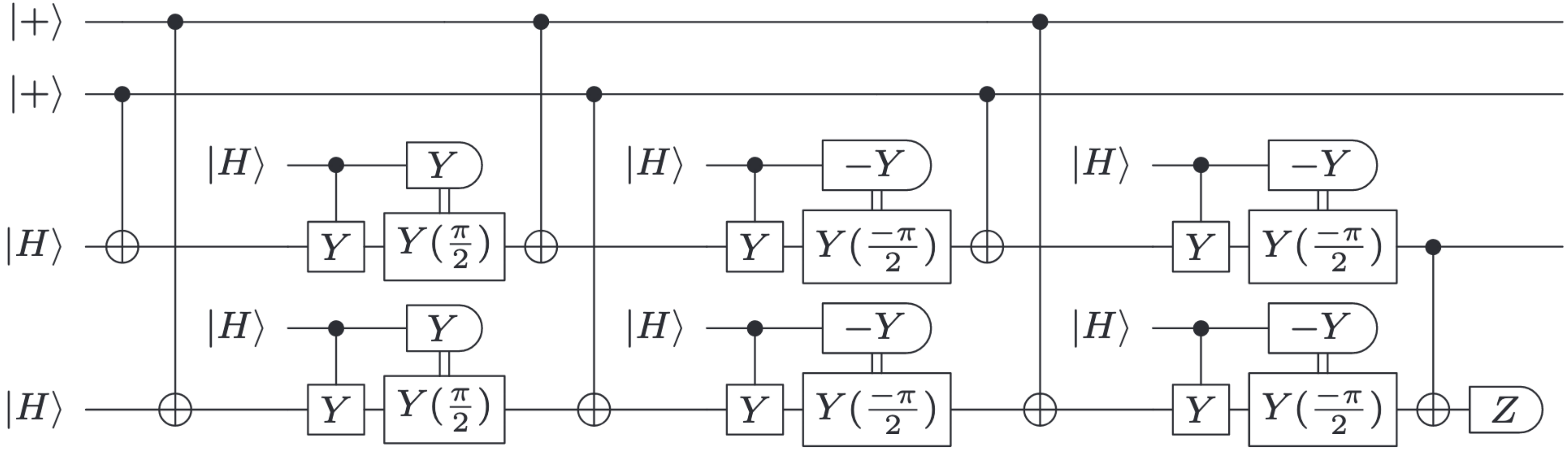


FIG. 5. Complete logical H-to-Toffoli distillation circuit.

Full Logical Distillation Circuit

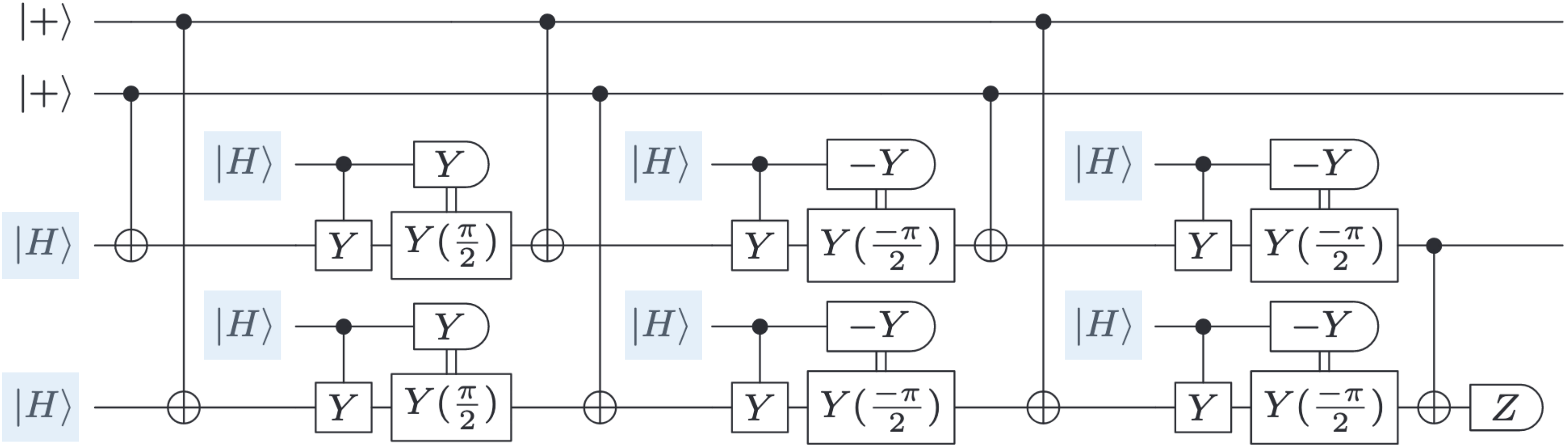


FIG. 5. Complete logical H-to-Toffoli distillation circuit.

Full Logical Distillation Circuit

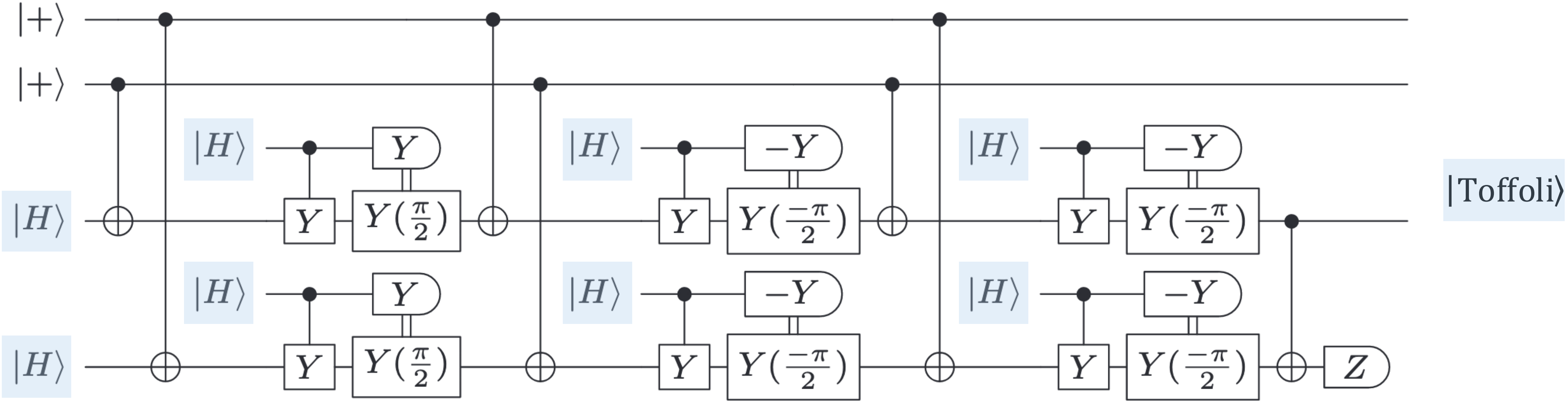


FIG. 5. Complete logical *H*-to-Toffoli distillation circuit.

Full Logical Distillation Circuit

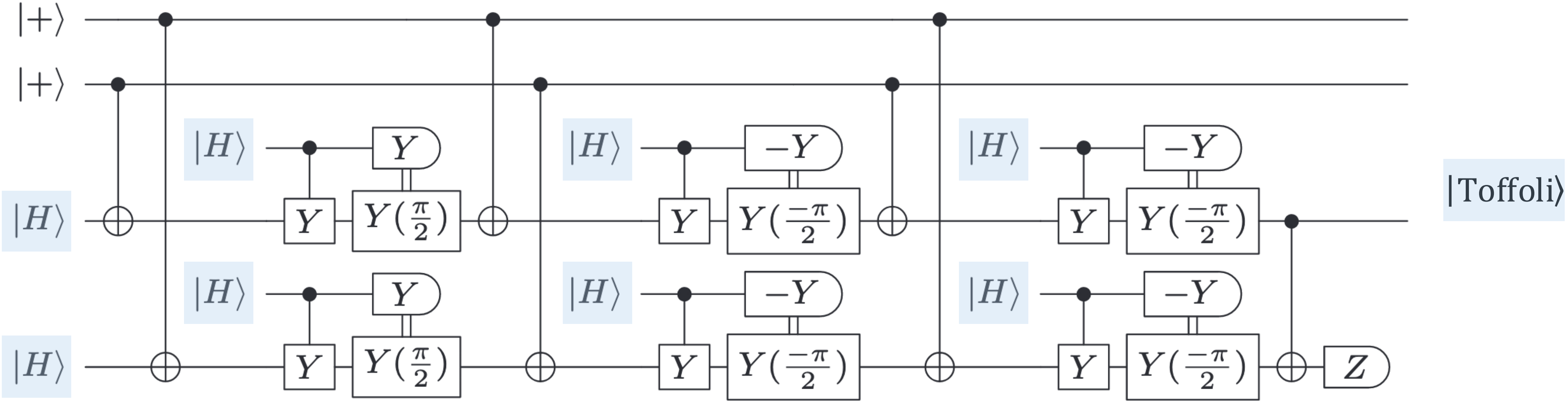


FIG. 5. Complete logical *H*-to-Toffoli distillation circuit.

Outputs $|\text{Toffoli}\rangle = (|000\rangle + |100\rangle + |010\rangle + |111\rangle)/2$
that can be injected to realize a logical Toffoli gate

Full Logical Distillation Circuit

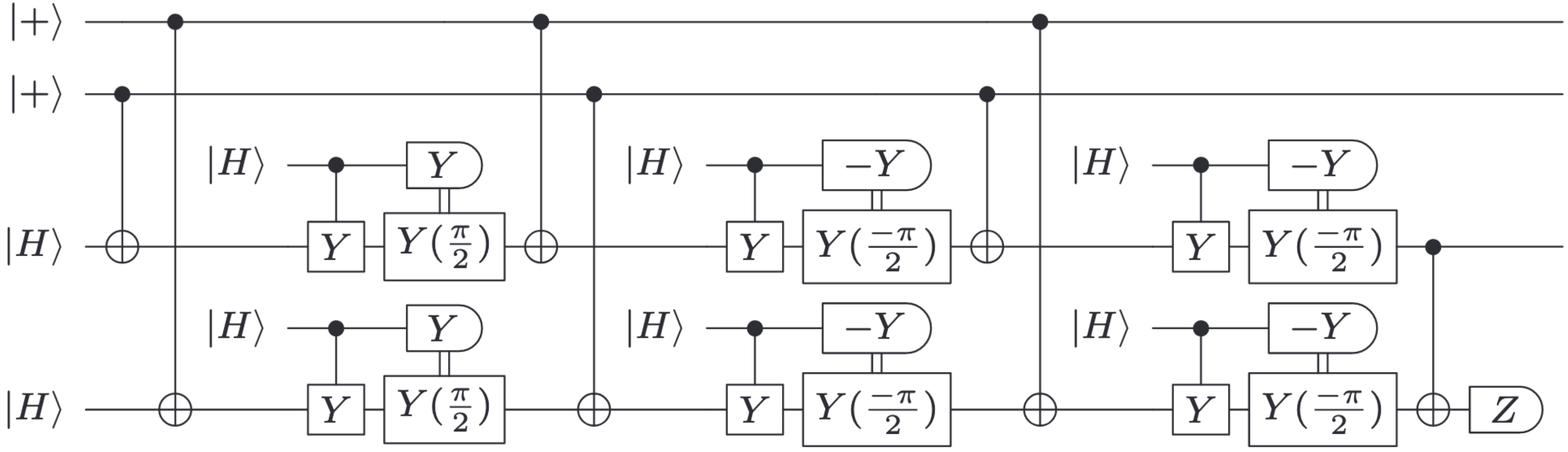
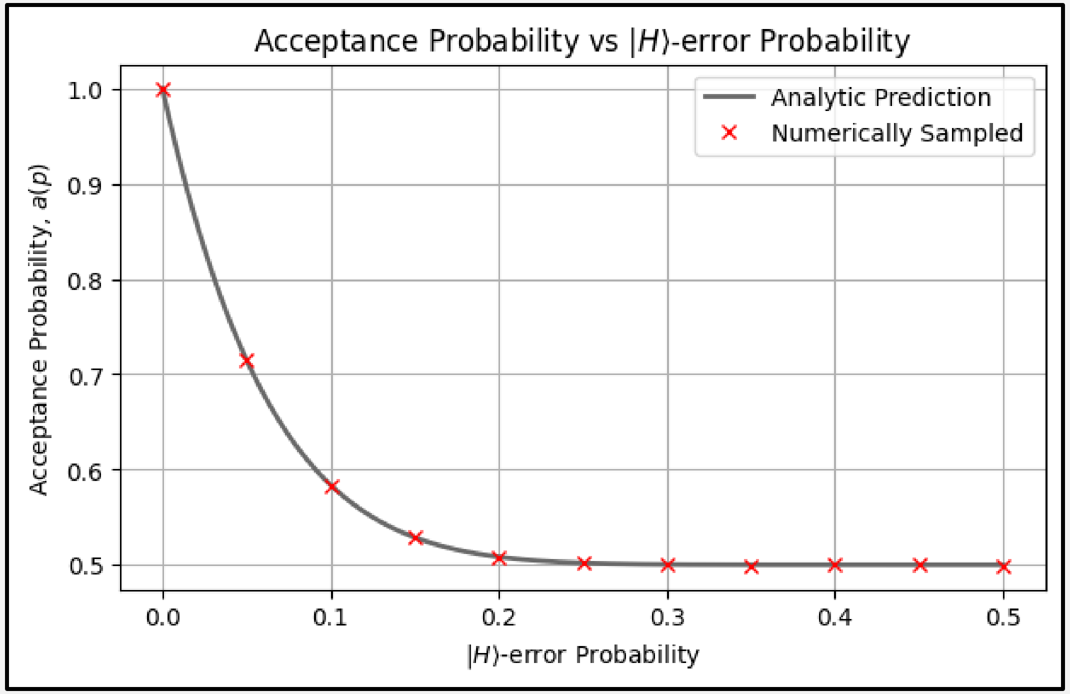


FIG. 5. Complete logical *H*-to-Toffoli distillation circuit.

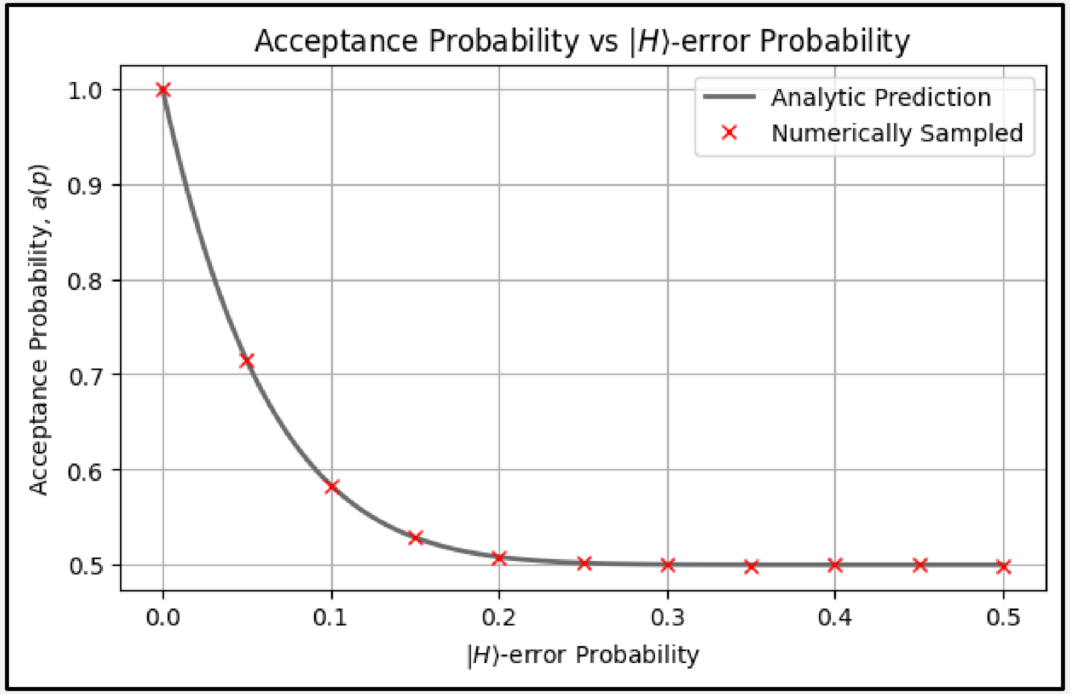
Reproducing $a(p)$, $e(p)a(p)$, for the Logical Circuit

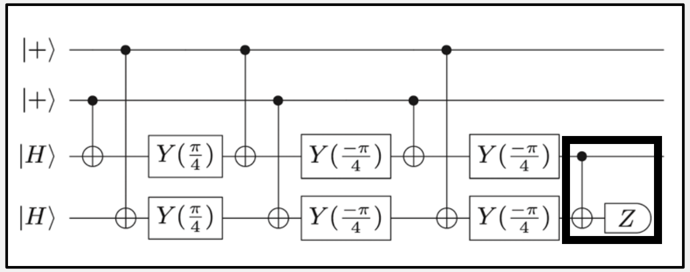
Assuming perfect Cliffords!



Reproducing $a(p)$, $e(p)a(p)$, for the Logical Circuit

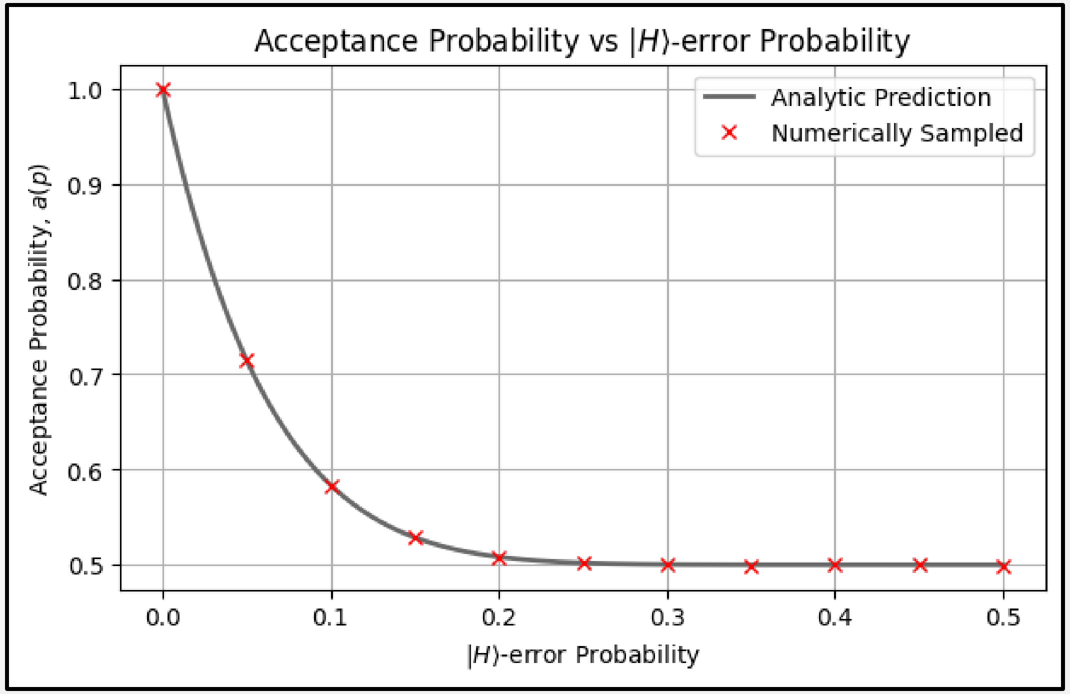
Assuming perfect Cliffords!

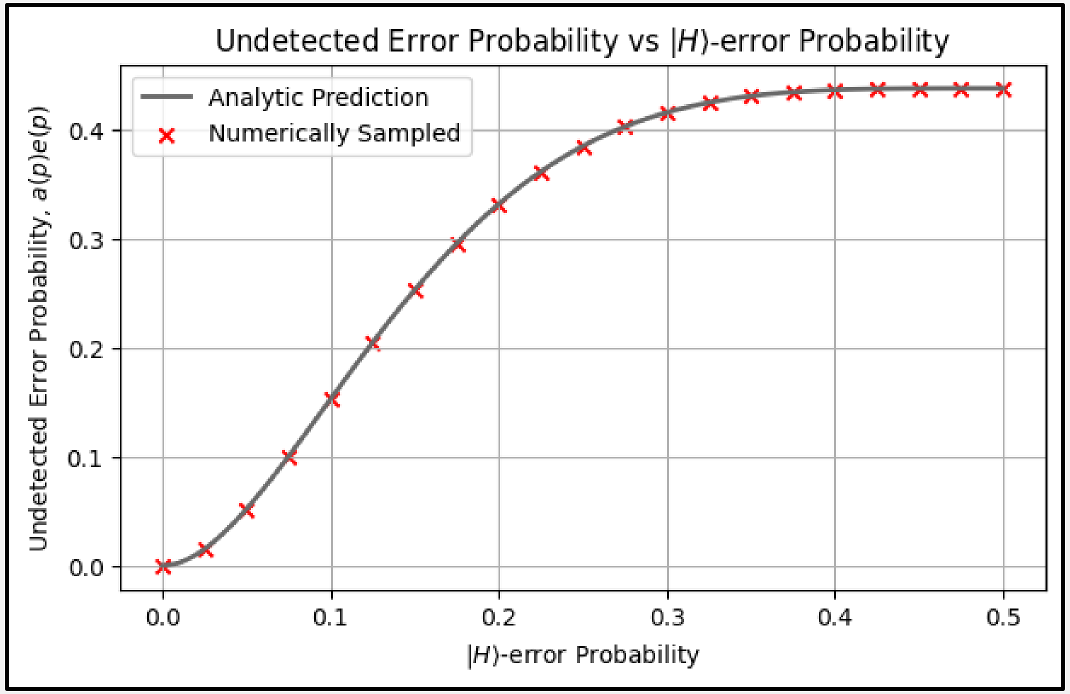


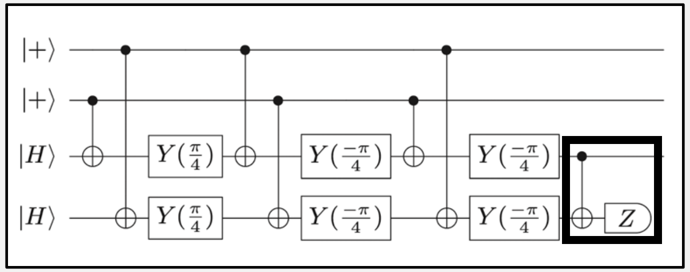


Reproducing $a(p)$, $e(p)a(p)$, for the Logical Circuit

Assuming perfect Cliffords!

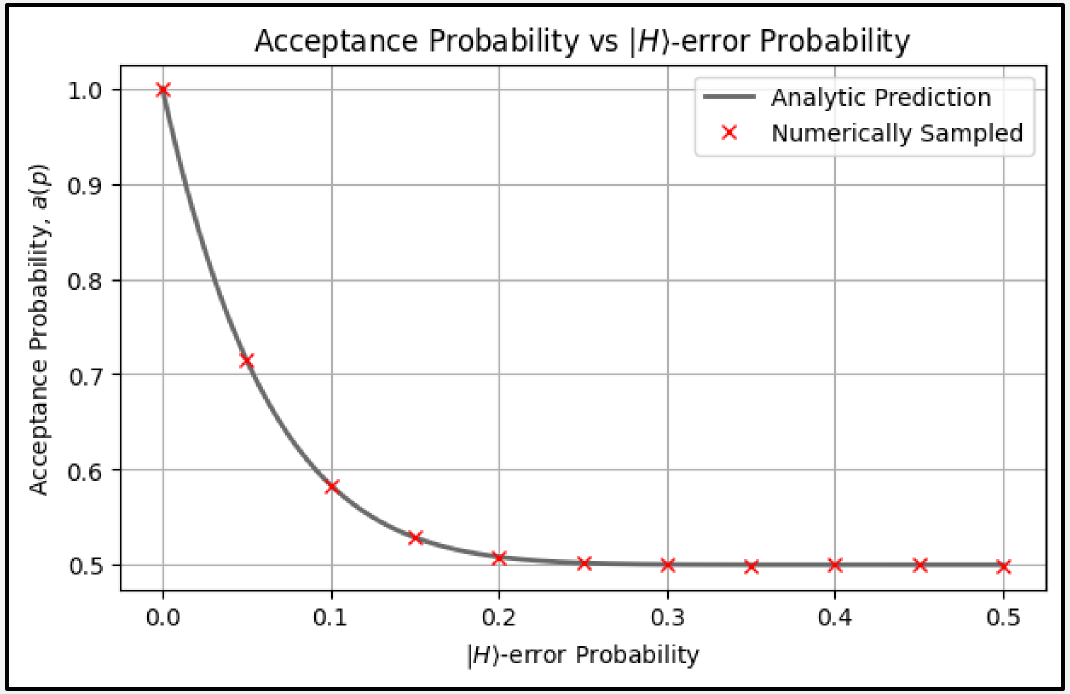


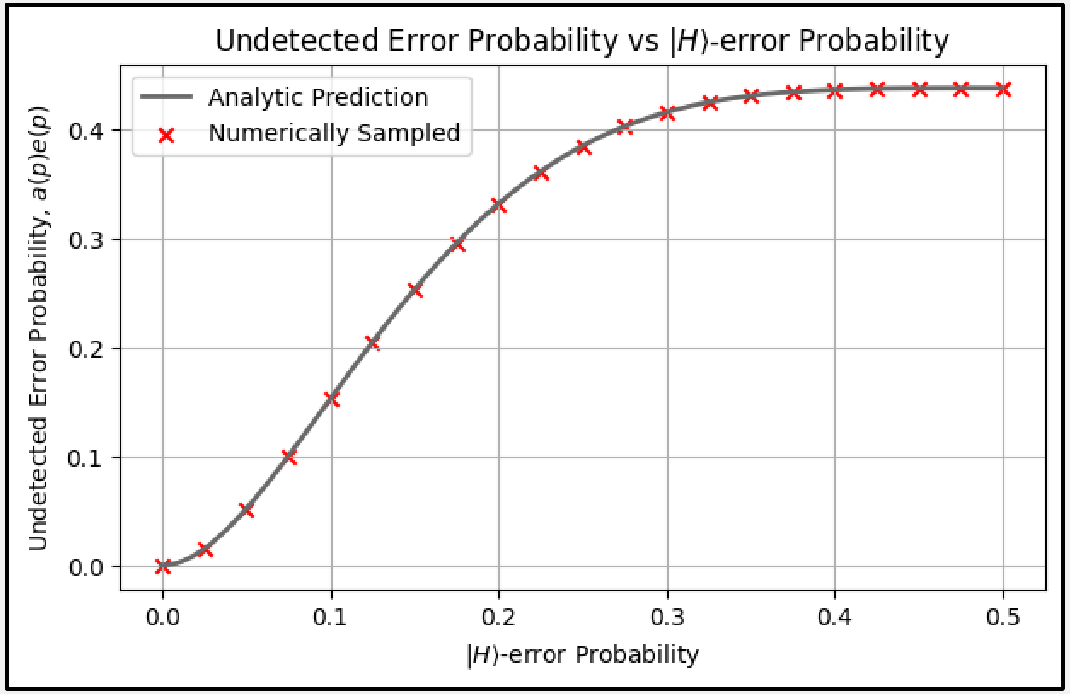


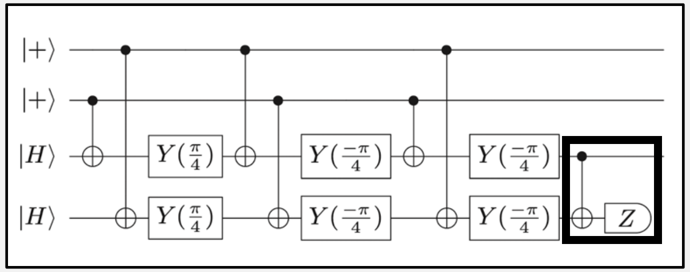


Reproducing $a(p)$, $e(p)a(p)$, for the Logical Circuit

Assuming perfect Cliffords!



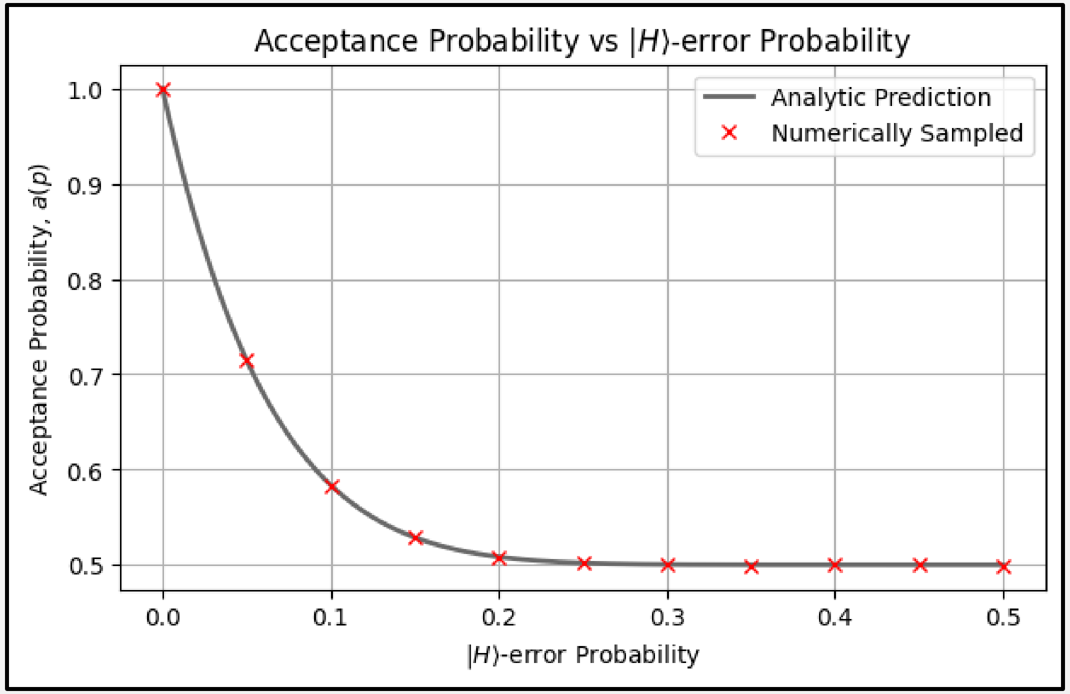


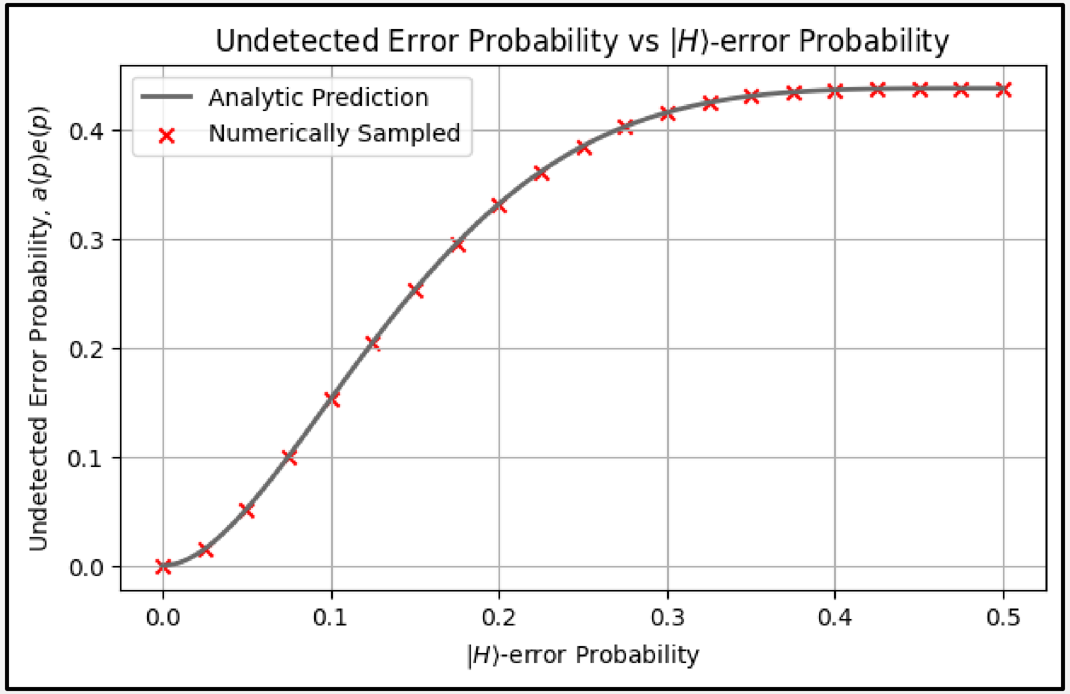


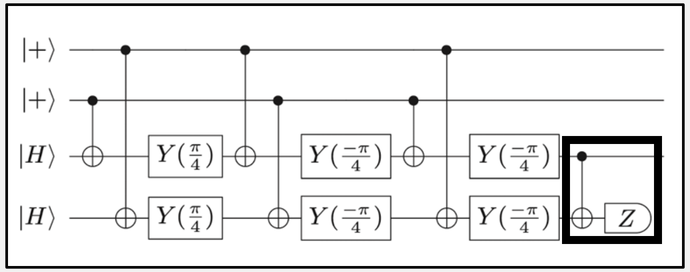
Confirms our circuit is implemented correctly, but...

Reproducing $a(p)$, $e(p)a(p)$, for the Logical Circuit

Assuming perfect Cliffords!



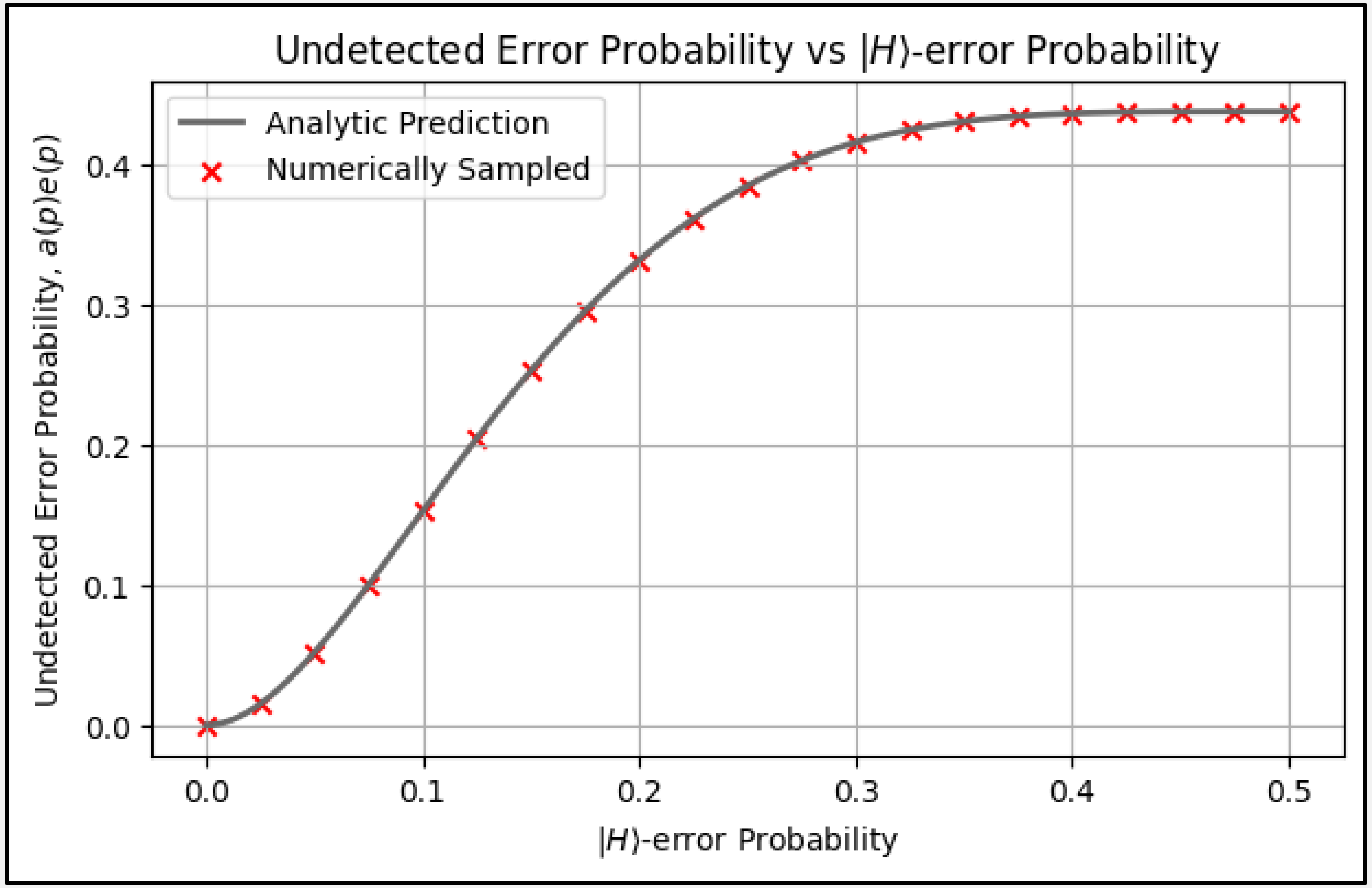




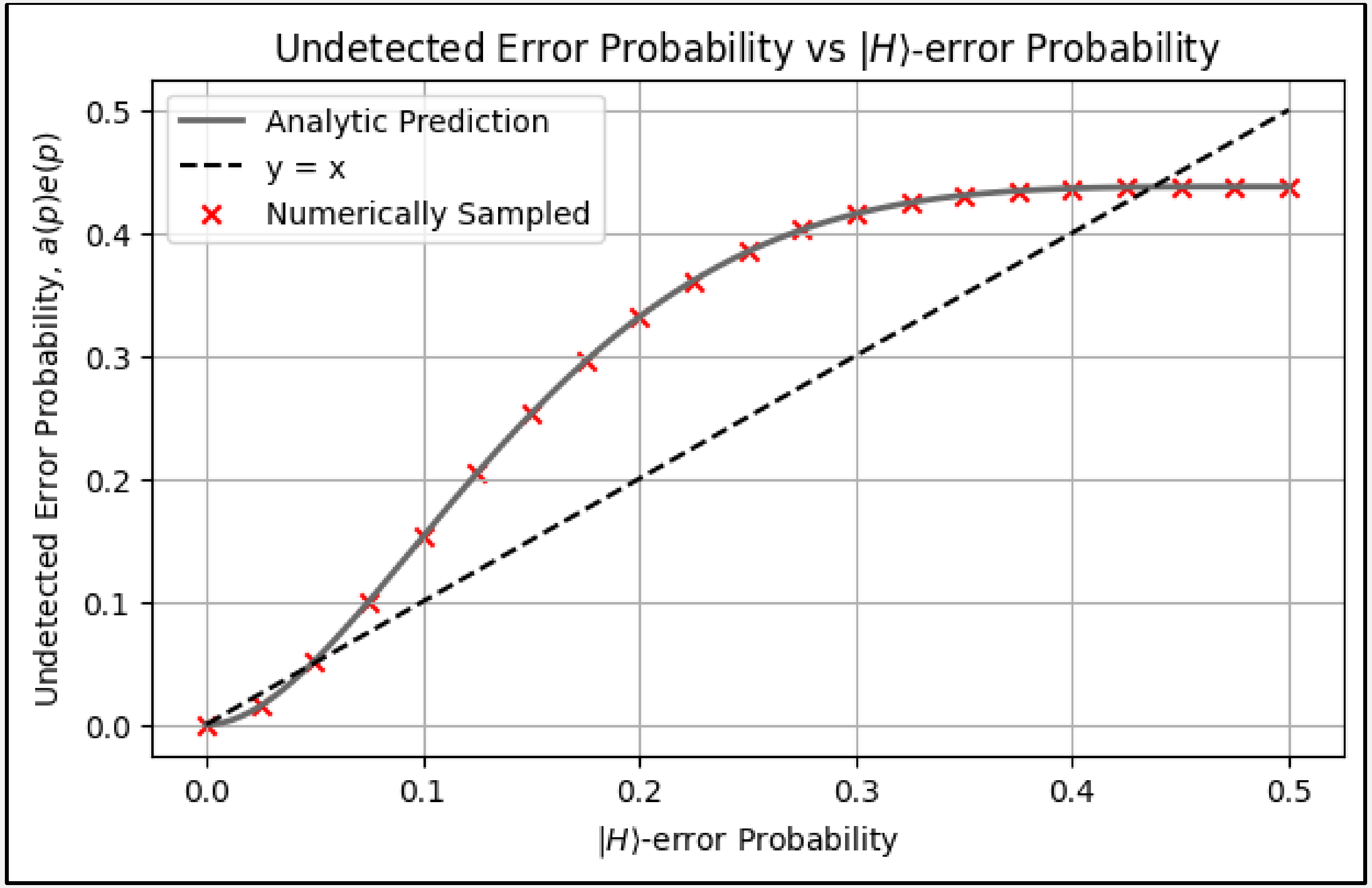
Confirms our circuit is implemented correctly, but...

when is it useful?

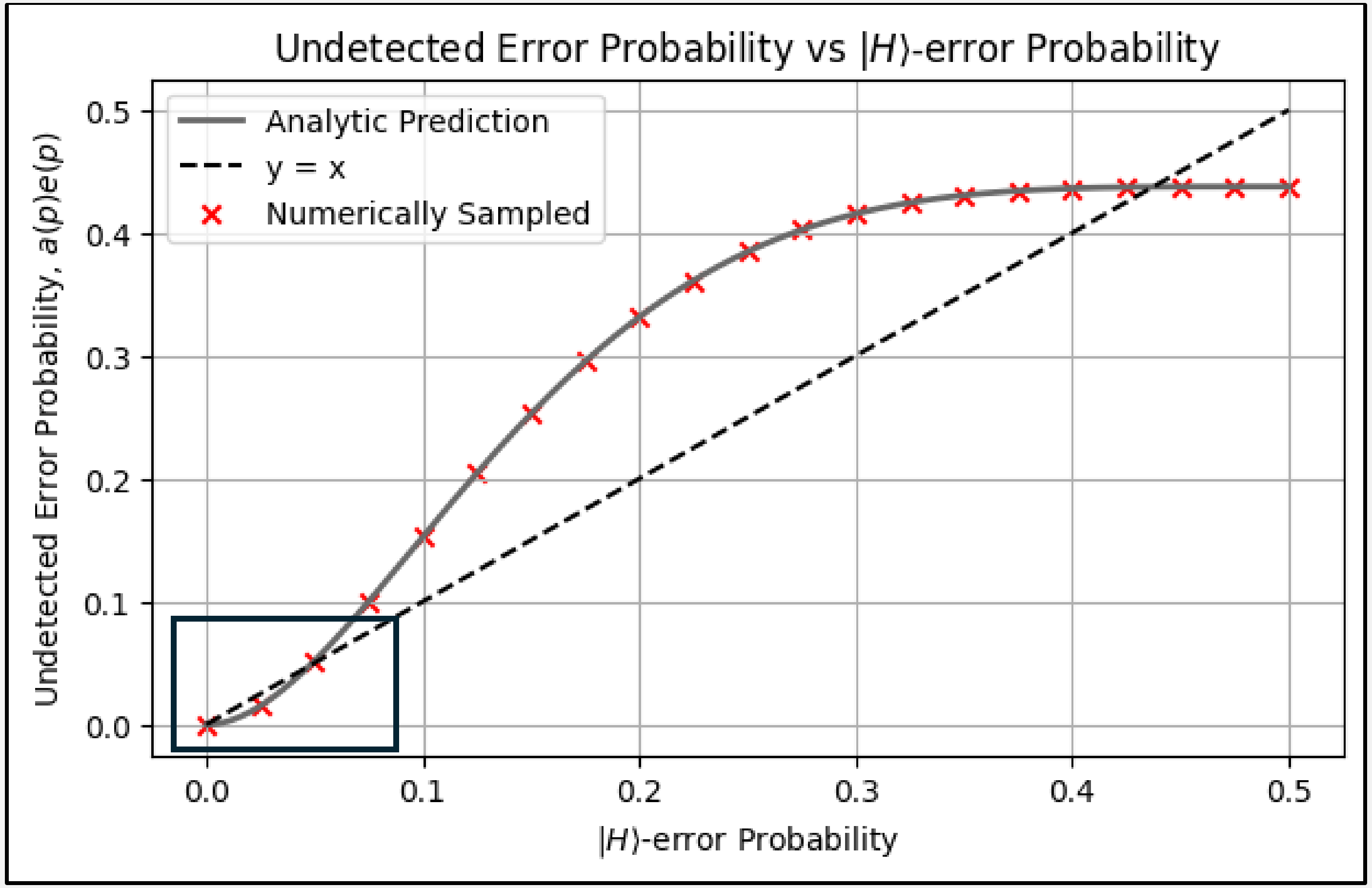
Undetected Error Probability, a Closer Examination



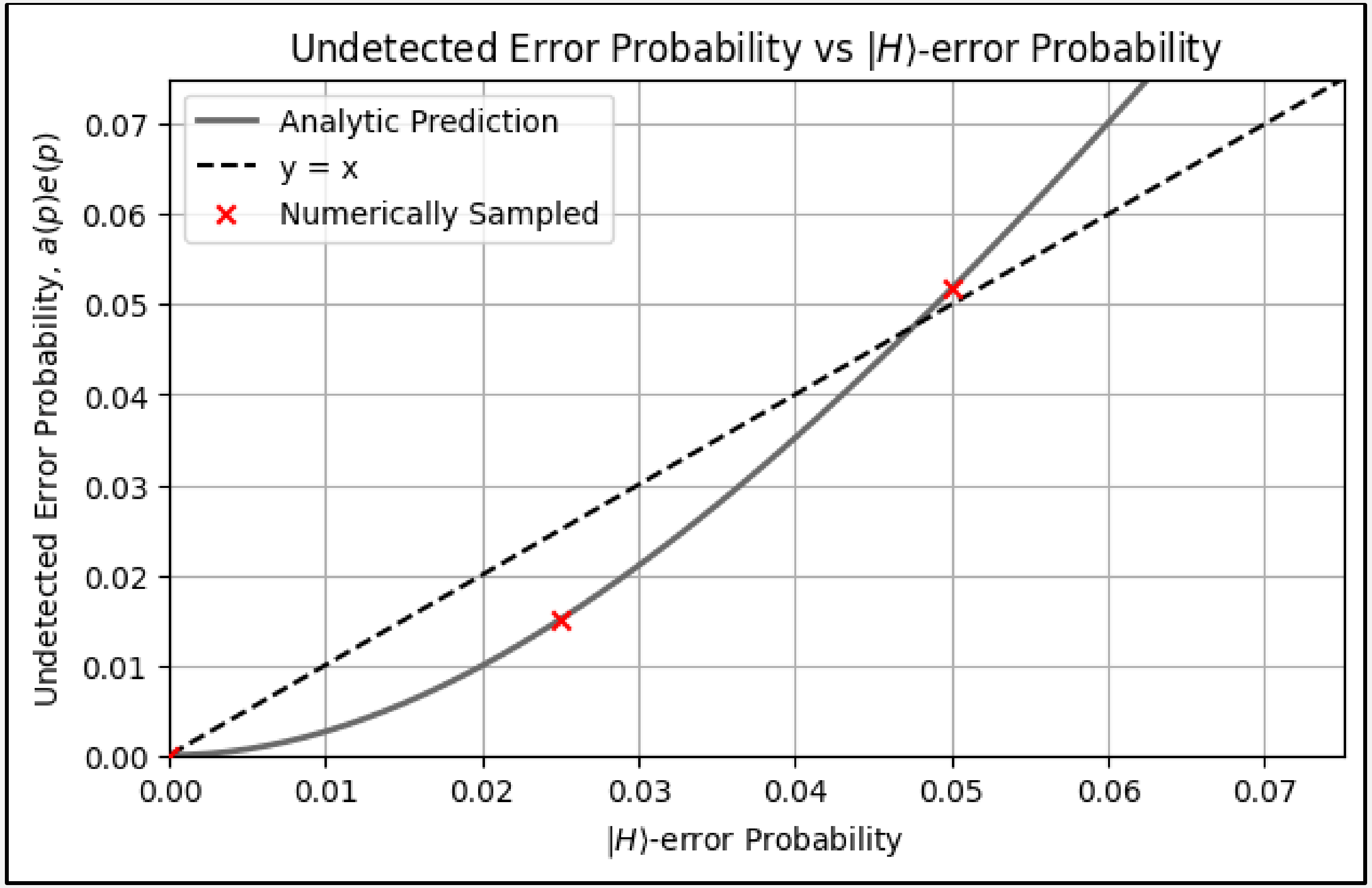
Undetected Error Probability, a Closer Examination



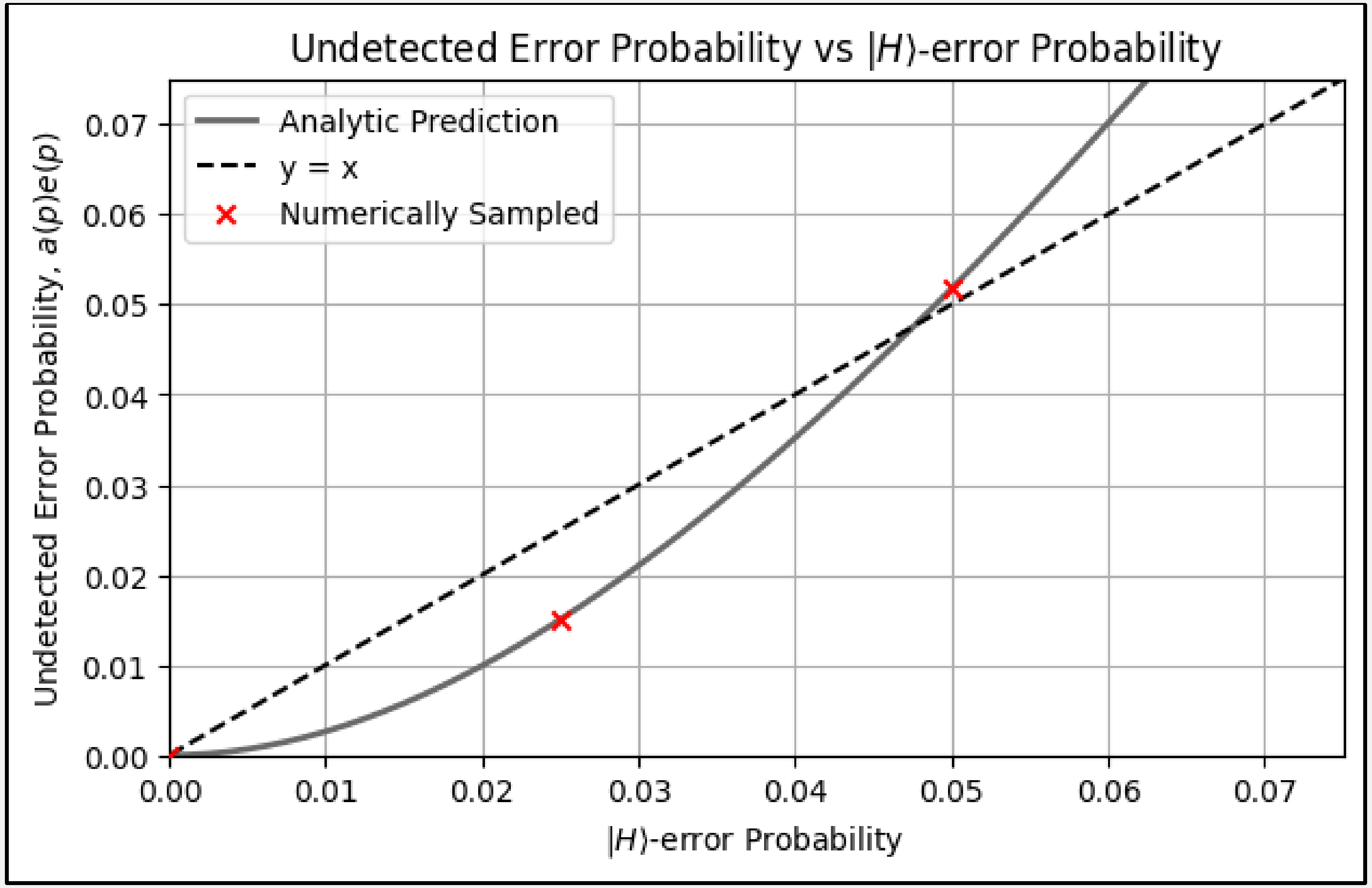
Undetected Error Probability, a Closer Examination



Undetected Error Probability, a Closer Examination



Undetected Error Probability, a Closer Examination



More on this later

Recall Research Question and Outline

Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Fully encoded circuit with base code

Recall Research Question and Outline

Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Fully encoded circuit with base code

1. Logical circuit
2. Encoding our logical circuit
3. Constructing our effective error channels

Recall Research Question and Outline

Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Fully encoded circuit with base code

1. ~~Logical circuit~~
2. Encoding our logical circuit
3. Constructing our effective error channels

Recall Research Question and Outline

Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Fully encoded circuit with base code

1. ~~Logical circuit~~
2. **Encoding our logical circuit**
3. Constructing our effective error channels

Recall Research Question and Outline

Logical circuit with effective error channels

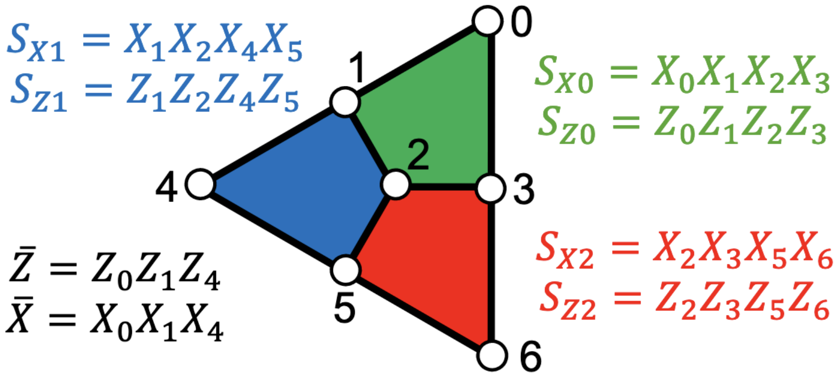
$$\overset{?}{=}$$

Fully encoded circuit with *STEANE CODE*

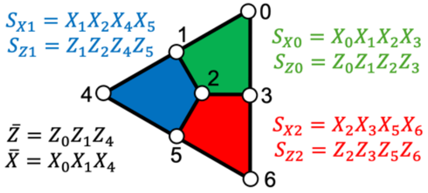
1. ~~Logical circuit~~
2. **Encoding our logical circuit**
3. Constructing our effective error channels

Steane-encoding

- The Steane Code
  - Detects and corrects all weight-1 errors
  - All Clifford gates can be implemented transversally



Full Steane-encoded Circuit



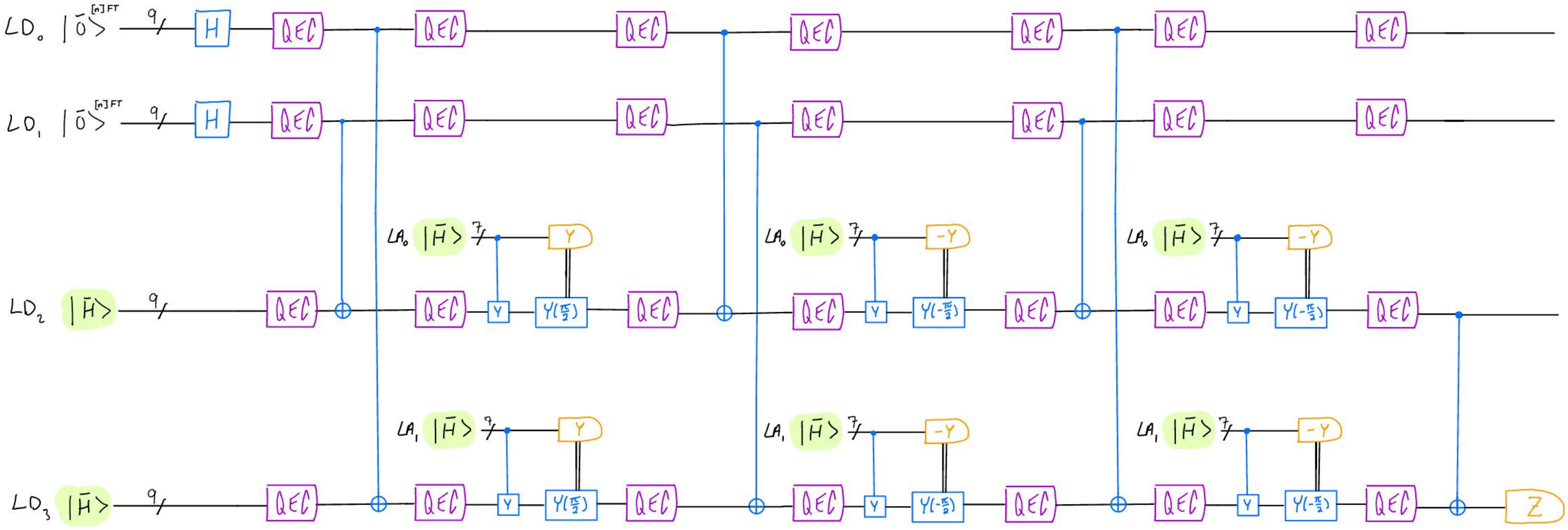


FIG. 6. Complete Steane-encoded $|H\rangle$ - $|\text{Toffoli}\rangle$ distillation circuit we want to simulate.

II. Effective Error Channels

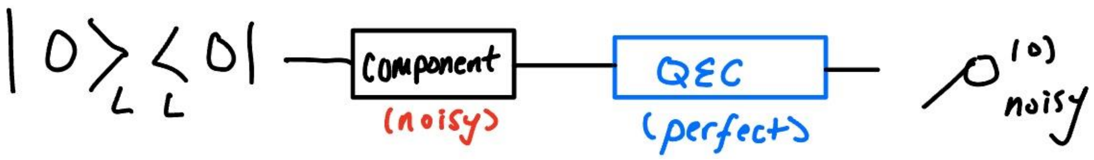
Process Tomography:

A method to reconstruct a full error channel describing the noise of a component under an arbitrary input state

One-Qubit Process Tomography

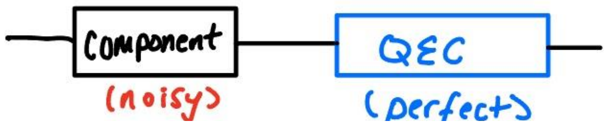
$|0\rangle_L \langle 0|$

One-Qubit Process Tomography



One-Qubit Process Tomography

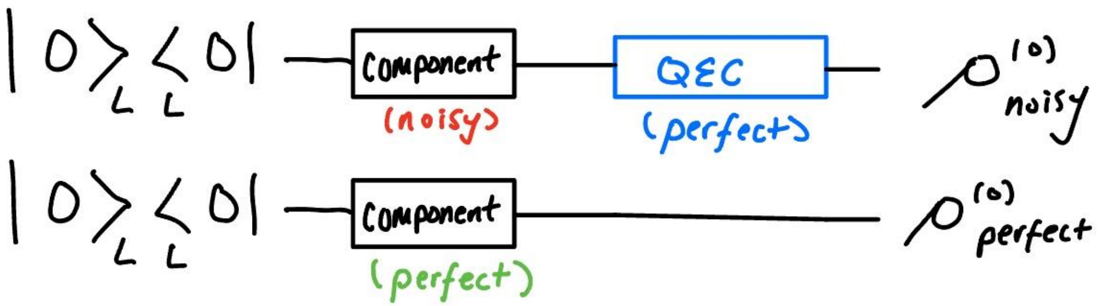
$|0\rangle_L \langle 0|_L$



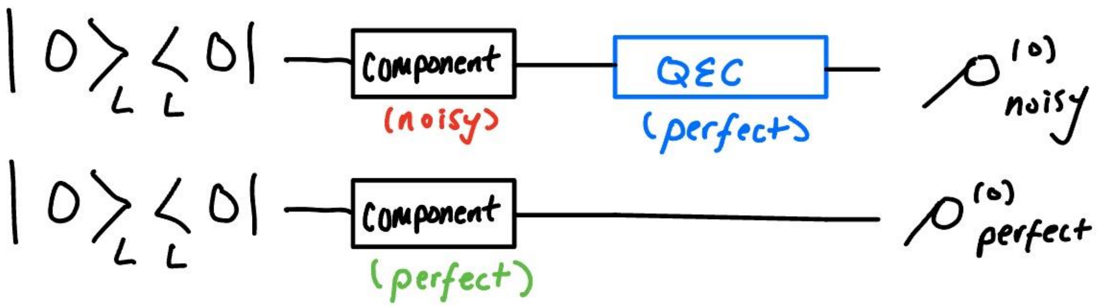
$\rho^{(0)}_{noisy}$

$|0\rangle_L \langle 0|_L$

One-Qubit Process Tomography

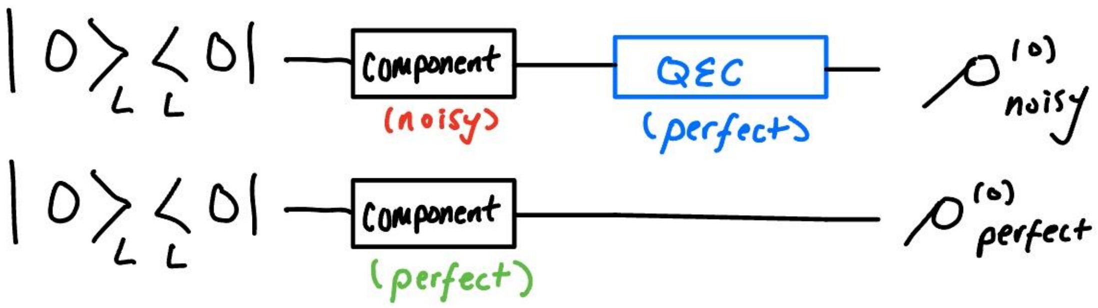


One-Qubit Process Tomography



$$1 - F\left[\rho^{(0)}_{noisy}, \rho^{(0)}_{perfect}\right] = p_x + p_y$$

One-Qubit Process Tomography



$$1 - F\left[\rho^{(0)}_{noisy}, \rho^{(0)}_{perfect}\right] = p_x + p_y$$

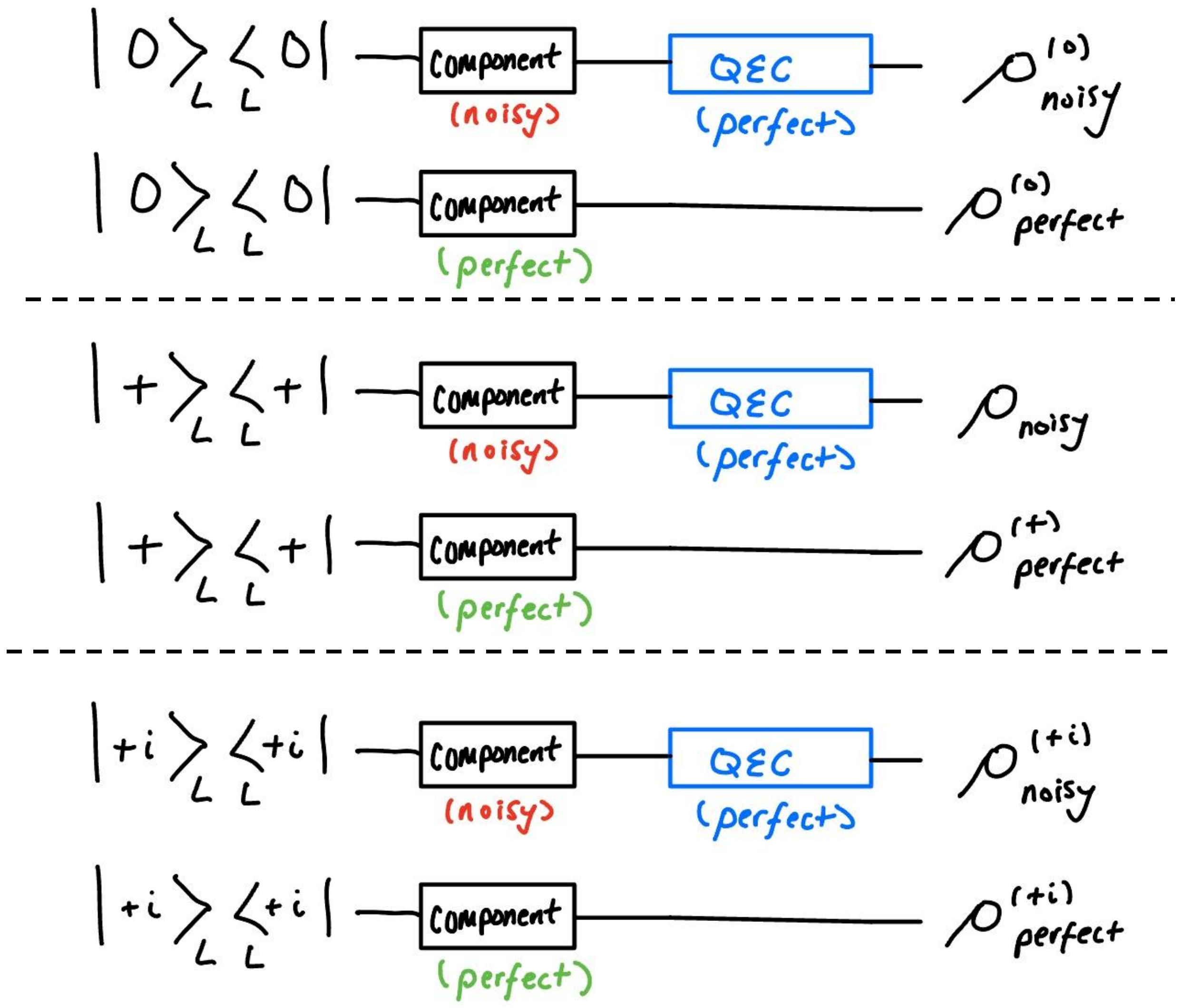
$|+\rangle_L\langle +|_L$

$|+\rangle_L\langle +|_L$

$|+i\rangle_L\langle +i|_L$

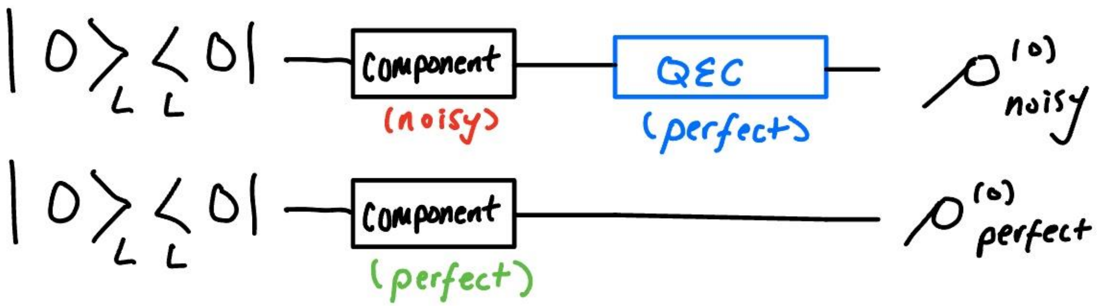
$|+i\rangle_L\langle +i|_L$

One-Qubit Process Tomography

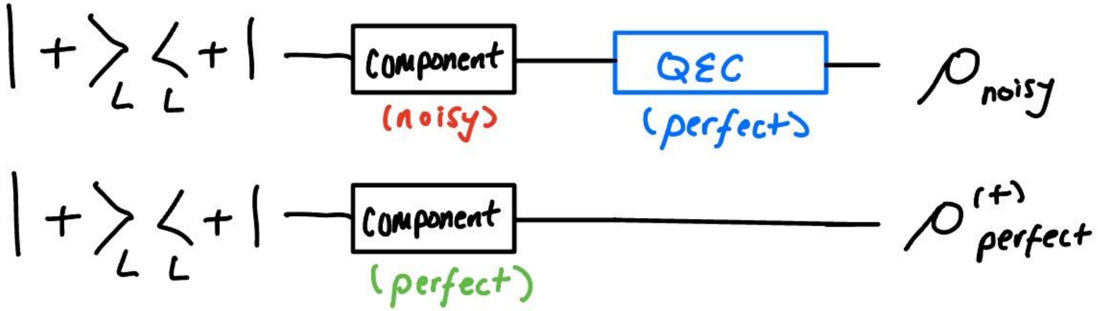


$$1 - F\left[\rho^{(0)}_{noisy}, \rho^{(0)}_{perfect}\right] = p_x + p_y$$

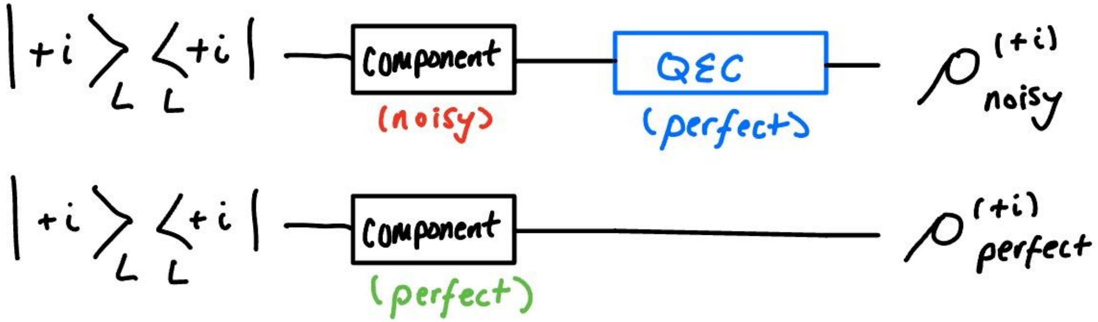
One-Qubit Process Tomography



$$1 - F\left[\rho^{(0)}_{noisy}, \rho^{(0)}_{perfect}\right] = p_x + p_y$$

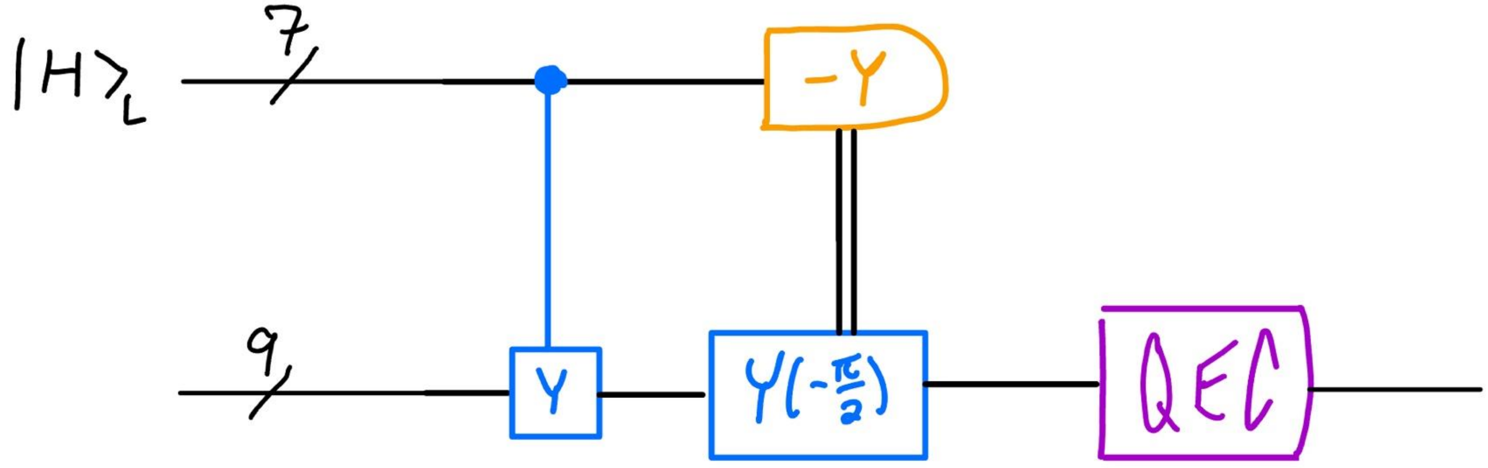


$$1 - F\left[\rho^{(+)}_{noisy}, \rho^{(+)}_{perfect}\right] = p_y + p_z$$

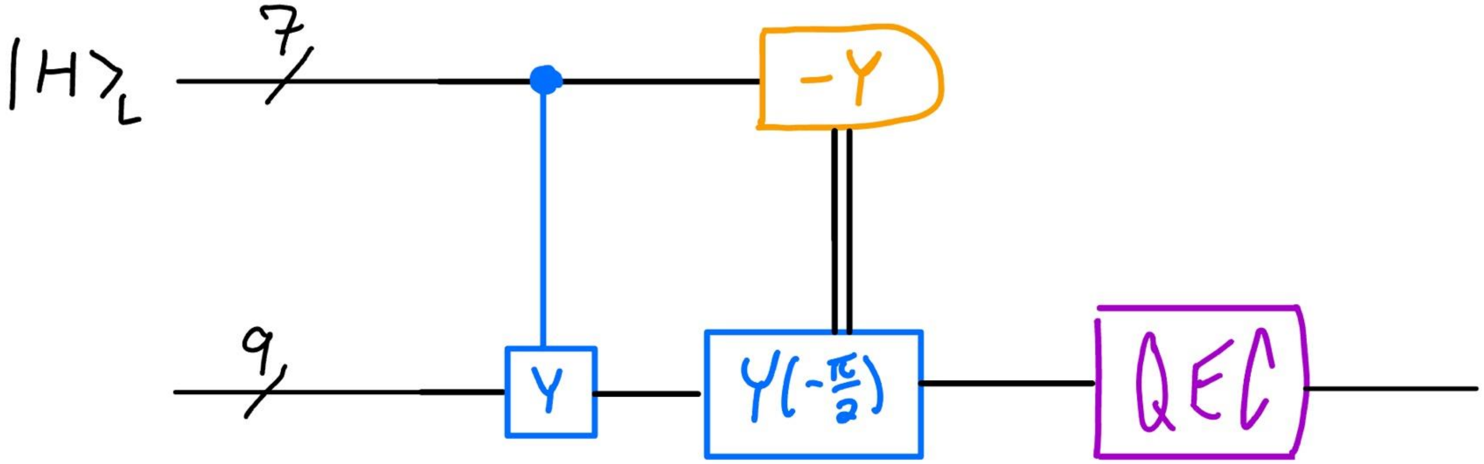


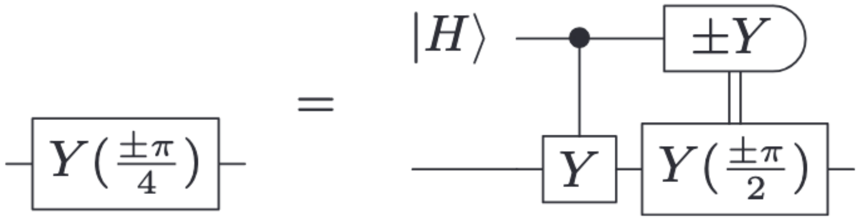
$$1 - F\left[\rho^{(+i)}_{noisy}, \rho^{(+i)}_{perfect}\right] = p_x + p_z$$

Logical Injection Circuit Tomography

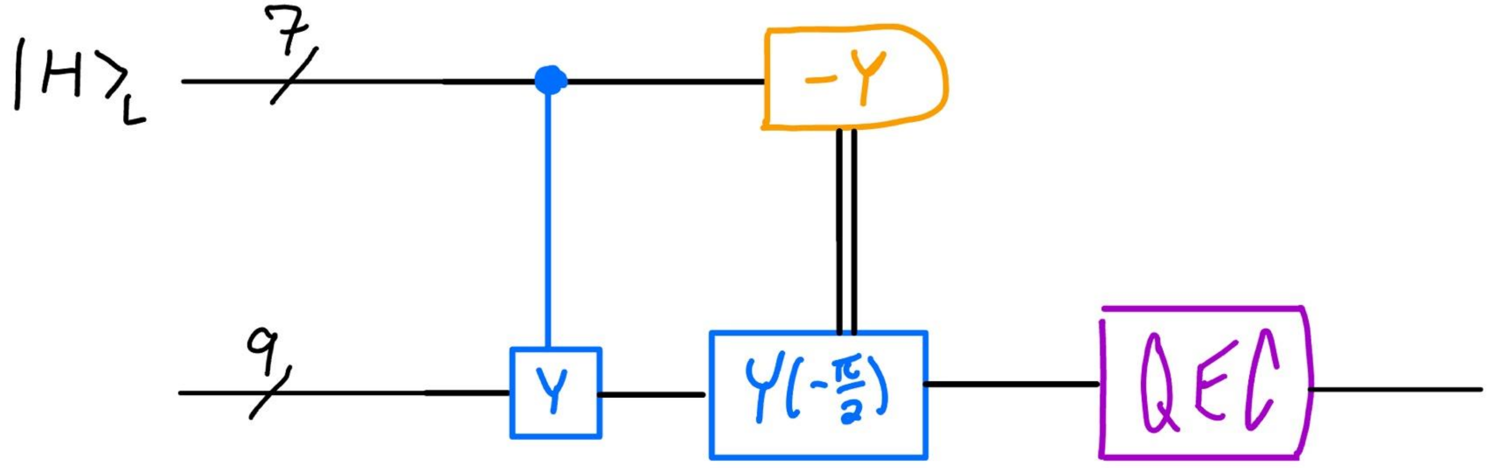


Logical Injection Circuit Tomography

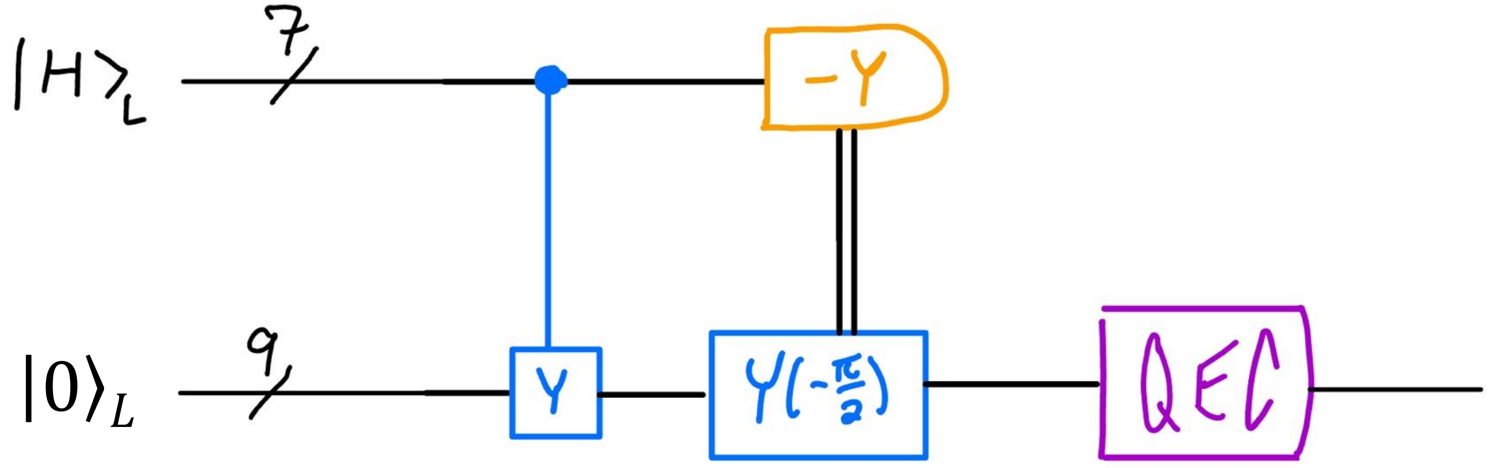




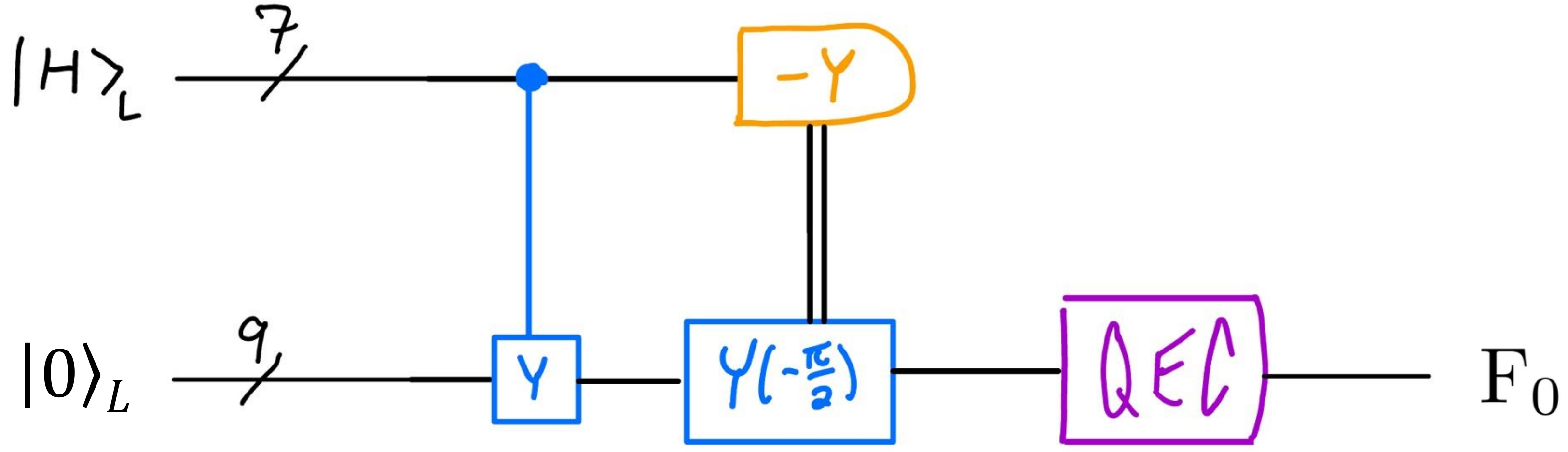
Logical Injection Circuit Tomography



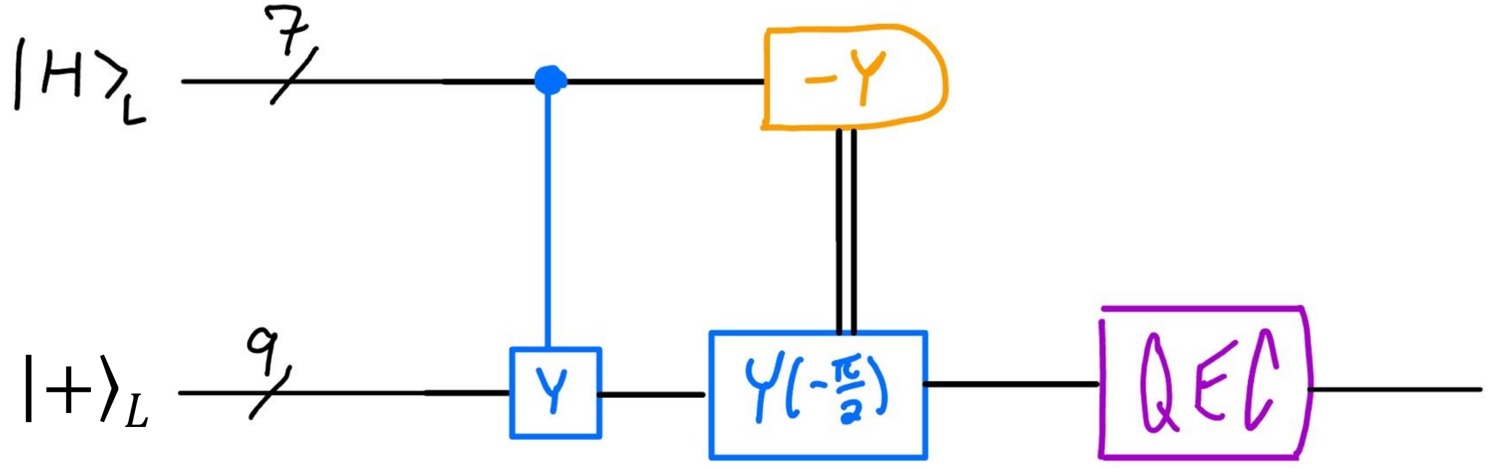
Logical Injection Circuit Tomography



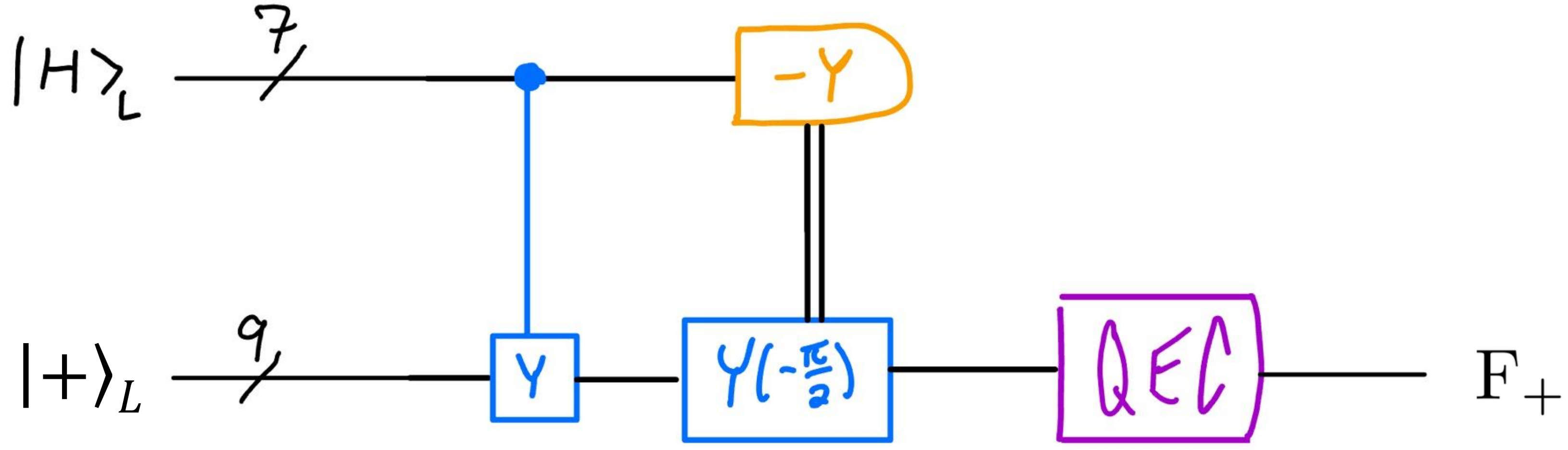
Logical Injection Circuit Tomography



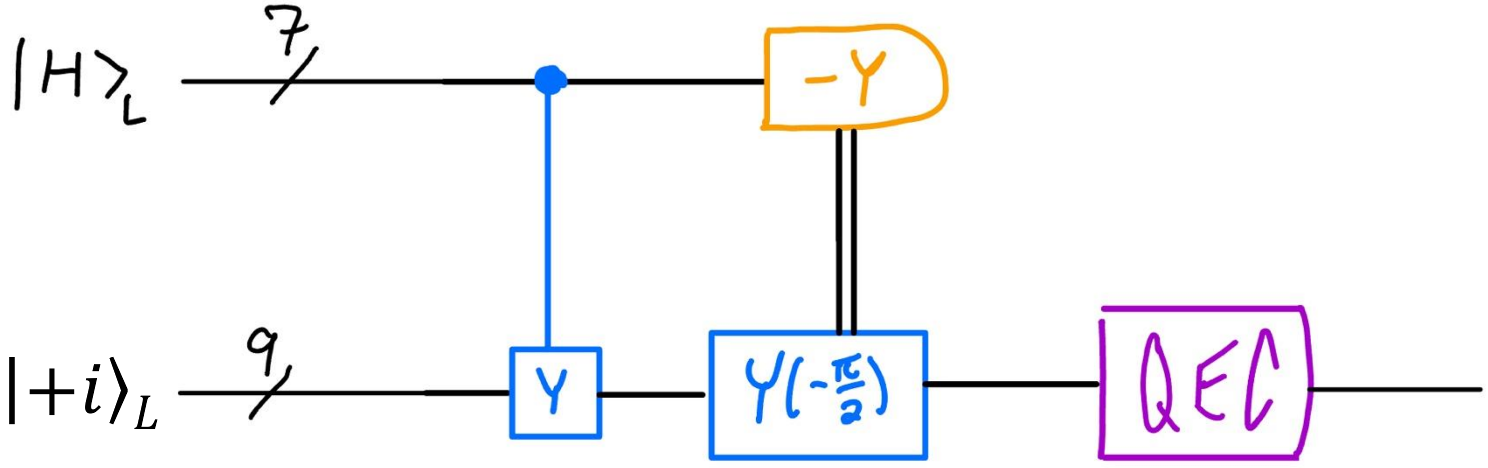
Logical Injection Circuit Tomography



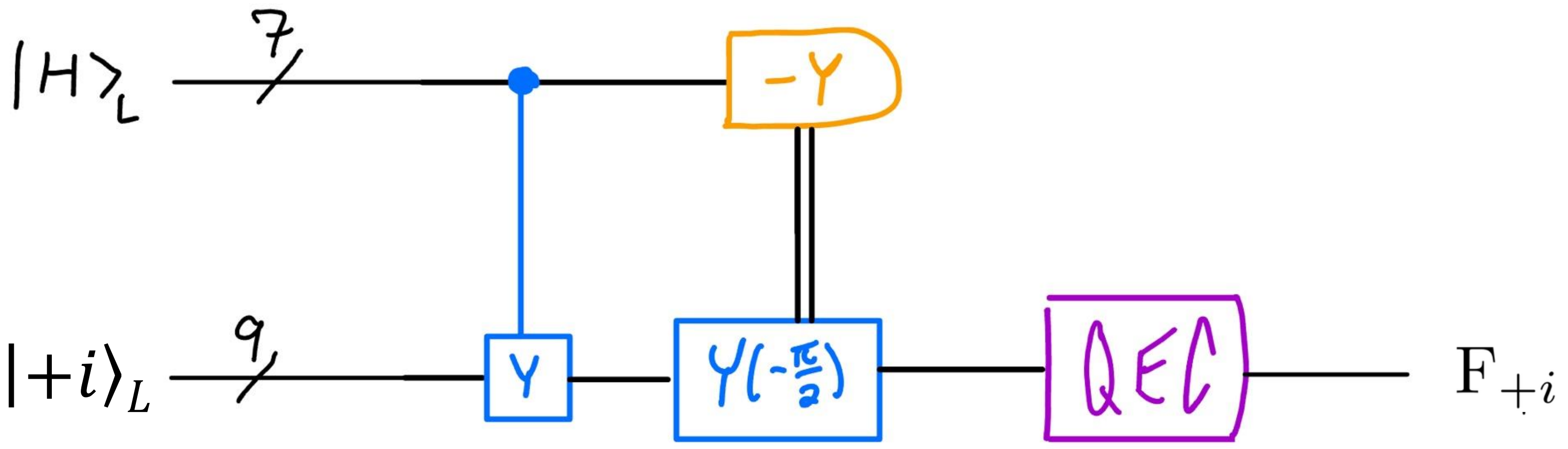
Logical Injection Circuit Tomography



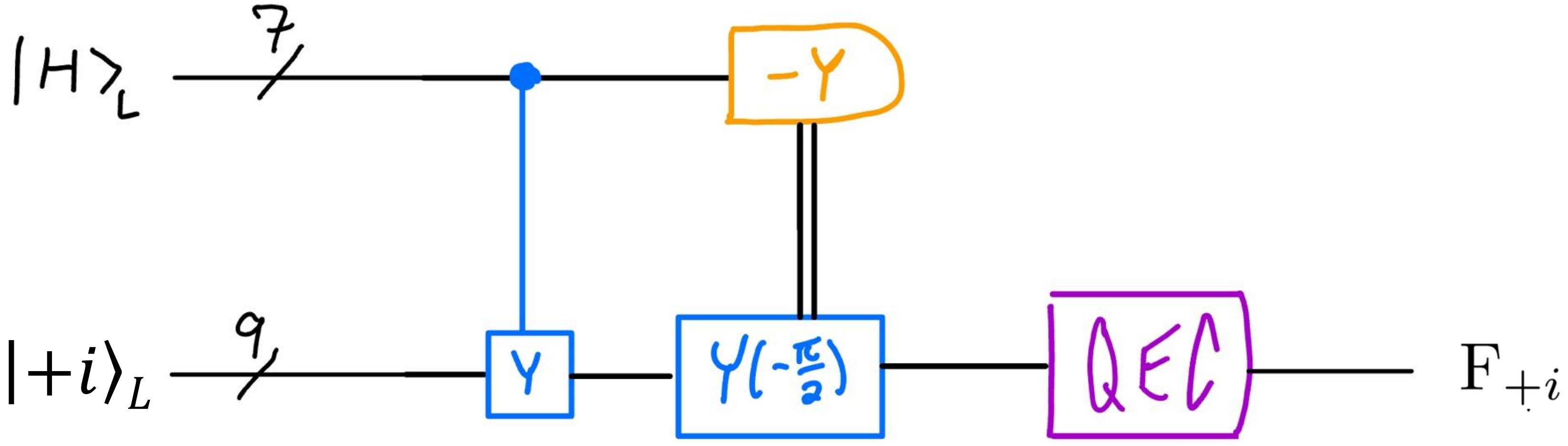
Logical Injection Circuit Tomography



Logical Injection Circuit Tomography



Logical Injection Circuit Tomography



solve system of equations, construct error channel

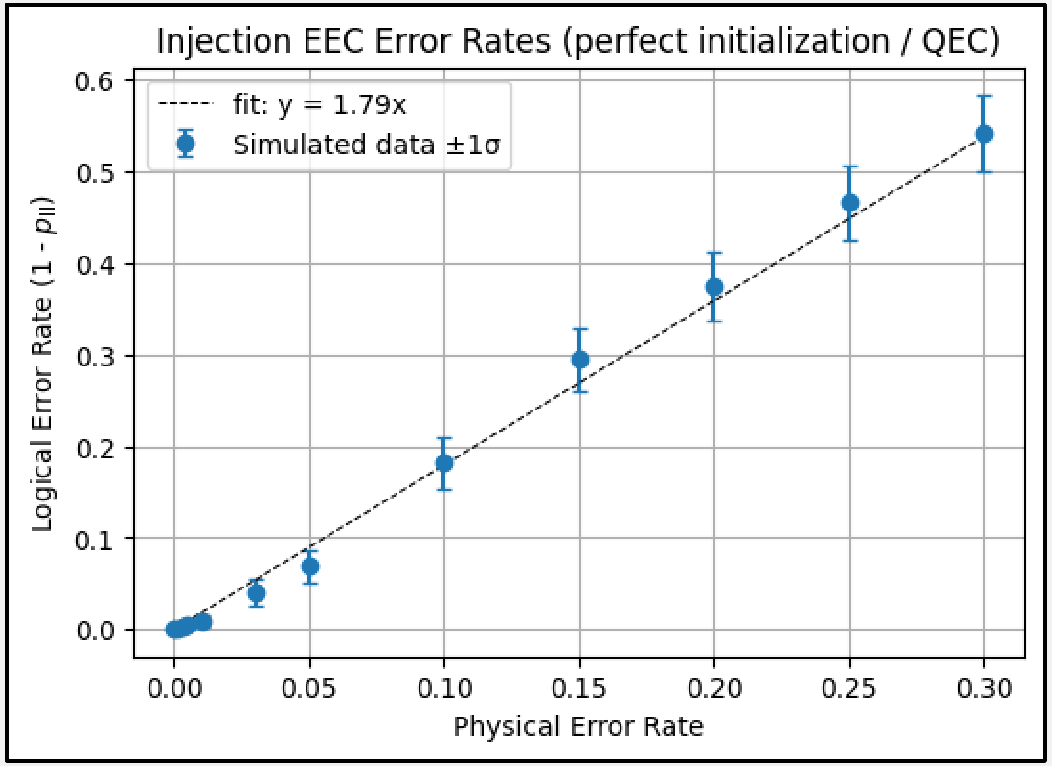
Two-Qubit Process Tomography *(thanks Aron)*

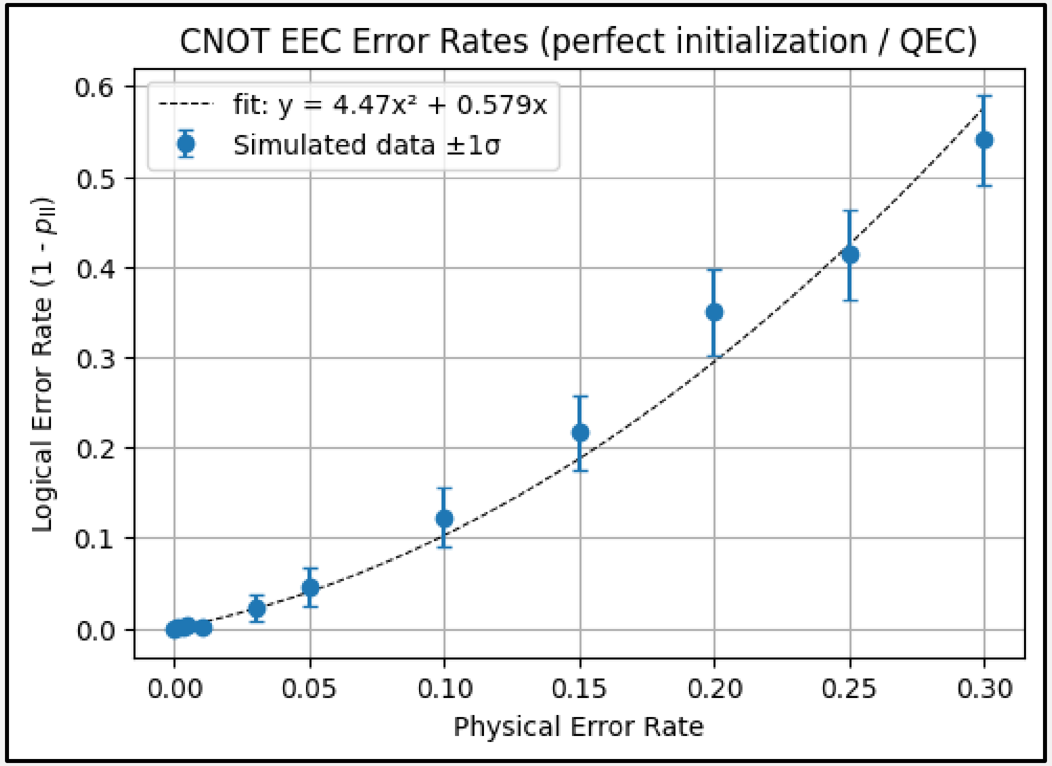
Initial state	Final measurements	Obtained probabilities
$\lvert\overline{00}\rangle$	$\bar{Z}_1,\ \bar{Z}_2$	$P^{ZZ}_{00}(-,+),\ P^{ZZ}_{00}(+,-),\ P^{ZZ}_{00}(-,-)$
$\lvert\overline{0+}\rangle$	$\bar{Z}_1,\ \bar{X}_2$	$P^{ZX}_{0+}(-,+),\ P^{ZX}_{0+}(+,-),\ P^{ZX}_{0+}(-,-)$
$\left\lvert\overline{0i}\right\rangle$	$\bar{Z}_1,\ \bar{Y}_2$	$P^{ZY}_{0i}(-,+),\ P^{ZY}_{0i}(+,-),\ P^{ZY}_{0i}(-,-)$
$\left\lvert\overline{i+}\right\rangle$	$\bar{Y}_1,\ \bar{X}_2$	$P^{YX}_{i+}(-,+),\ P^{YX}_{i+}(+,-),\ P^{YX}_{i+}(-,-)$
$\lvert\overline{++}\rangle$	$\bar{X}_1,\ \bar{X}_2$	$P^{XX}_{++}(-,+),\ P^{XX}_{++}(+,-),\ P^{XX}_{++}(-,-)$
$\lvert\overline{+0}\rangle$	$\bar{Y}_1\bar{Y}_2$	$P^{YY}_{+0}(-)$
$\left\lvert\overline{+i}\right\rangle$	$\bar{Y}_1\bar{Z}_2$	$P^{YZ}_{+i}(-)$
$\left\lvert\overline{i0}\right\rangle$	$\bar{X}_1\bar{Y}_2$	$P^{XY}_{i0}(-)$
$\left\lvert\overline{ii}\right\rangle$	$\bar{X}_1\bar{Z}_2$	$P^{XZ}_{ii}(-)$

Two-Qubit Process Tomography *(thanks Aron)*

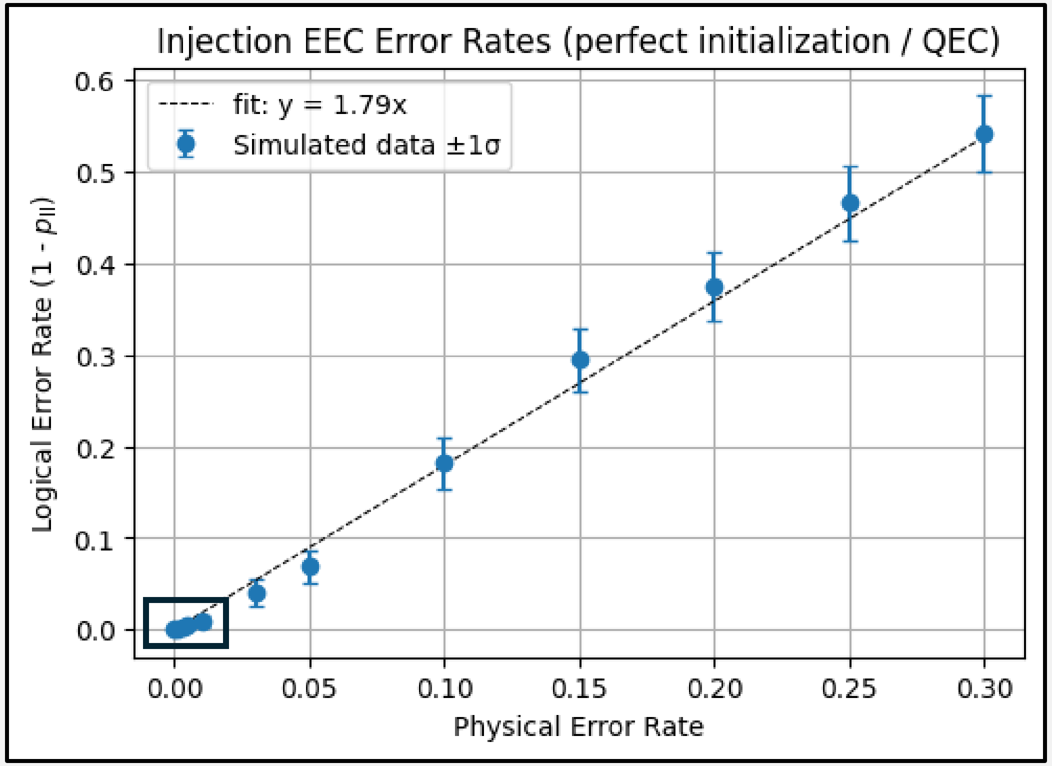
Initial state	Final measurements	Obtained probabilities
$\lvert\overline{00}\rangle$	$\bar{Z}_1,\ \bar{Z}_2$	$P_{00}^{ZZ}(-,+),\ P_{00}^{ZZ}(+,-),\ P_{00}^{ZZ}(-,-)$
$\lvert\overline{0+}\rangle$	$\bar{Z}_1,\ \bar{X}_2$	$P_{0+}^{ZX}(-,+),\ P_{0+}^{ZX}(+,-),\ P_{0+}^{ZX}(-,-)$
$\left\lvert\overline{0i}\right\rangle$	$\bar{Z}_1,\ \bar{Y}_2$	$P_{0i}^{ZY}(-,+),\ P_{0i}^{ZY}(+,-),\ P_{0i}^{ZY}(-,-)$
$\left\lvert\overline{i+}\right\rangle$	$\bar{Y}_1,\ \bar{X}_2$	$P_{i+}^{YX}(-,+),\ P_{i+}^{YX}(+,-),\ P_{i+}^{YX}(-,-)$
$\lvert\overline{++}\rangle$	$\bar{X}_1,\ \bar{X}_2$	$P_{++}^{XX}(-,+),\ P_{++}^{XX}(+,-),\ P_{++}^{XX}(-,-)$
$\lvert\overline{+0}\rangle$	$\bar{Y}_1\bar{Y}_2$	$P_{+0}^{YY}(-)$
$\left\lvert\overline{+i}\right\rangle$	$\bar{Y}_1\bar{Z}_2$	$P_{+i}^{YZ}(-)$
$\left\lvert\overline{i0}\right\rangle$	$\bar{X}_1\bar{Y}_2$	$P_{i0}^{XY}(-)$
$\left\lvert\overline{ii}\right\rangle$	$\bar{X}_1\bar{Z}_2$	$P_{ii}^{XZ}(-)$

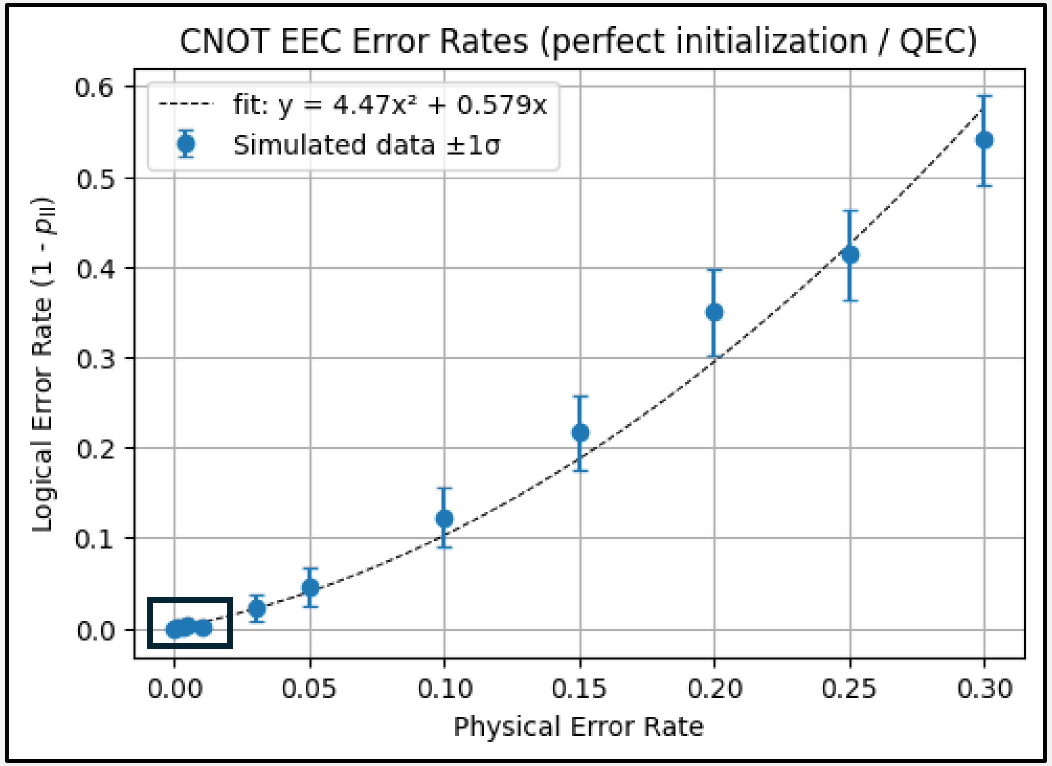
Effective Error Channels PER vs LER



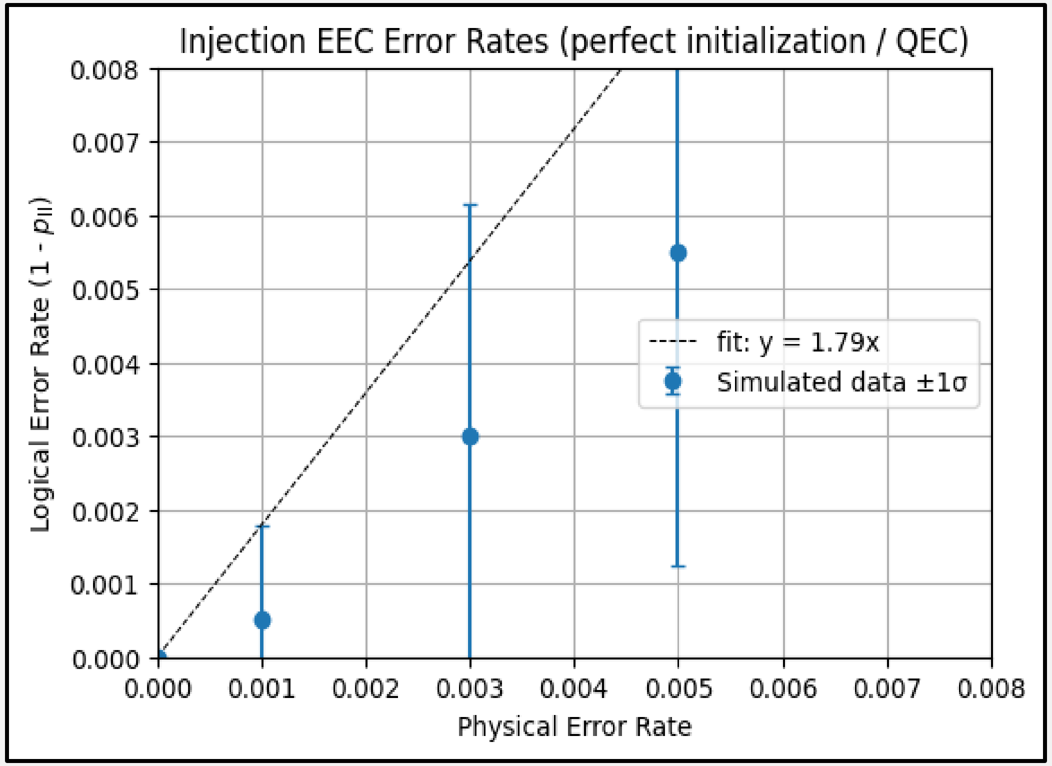


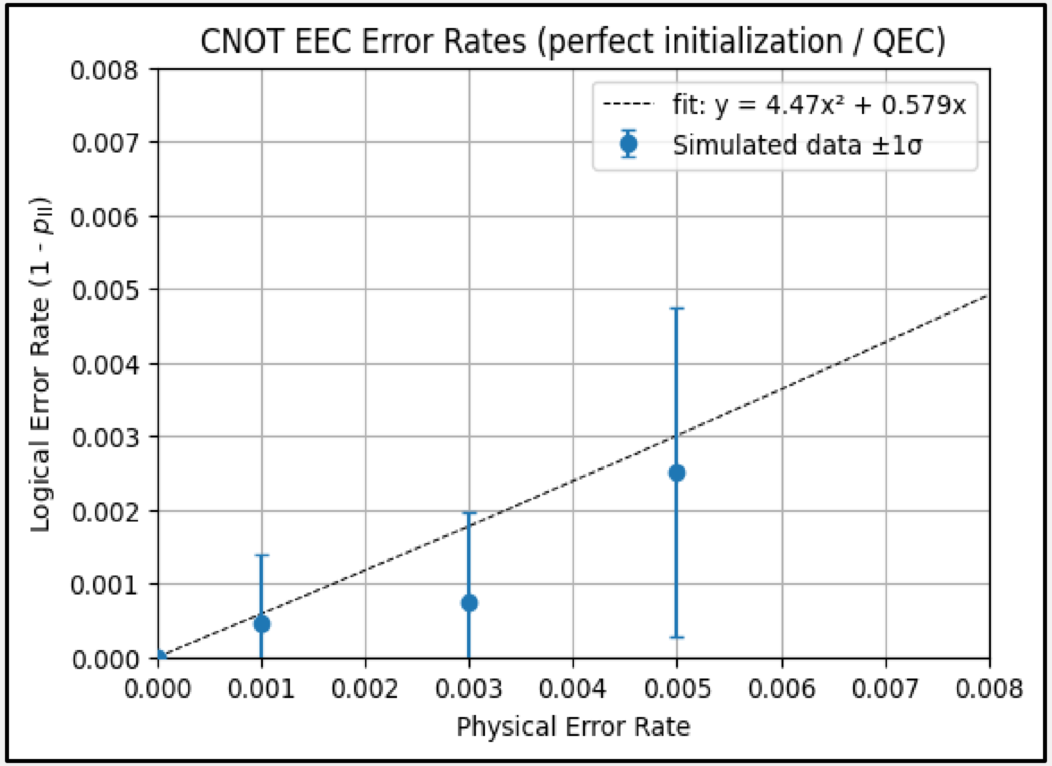
Effective Error Channels PER vs LER



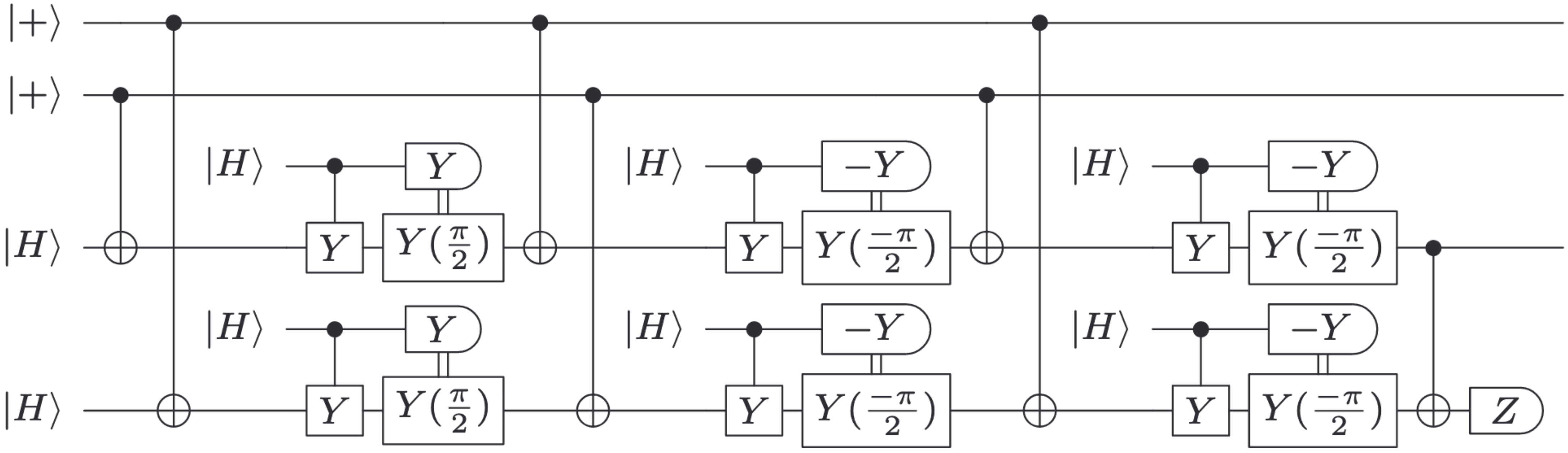


Effective Error Channels PER vs LER

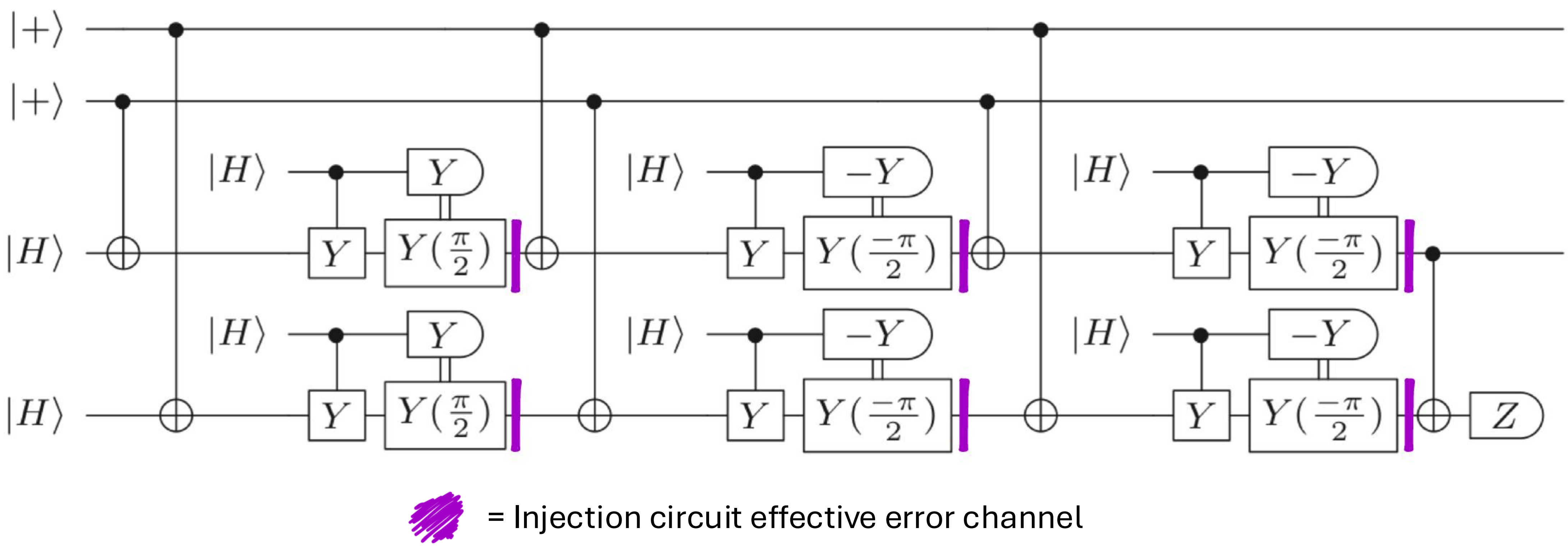




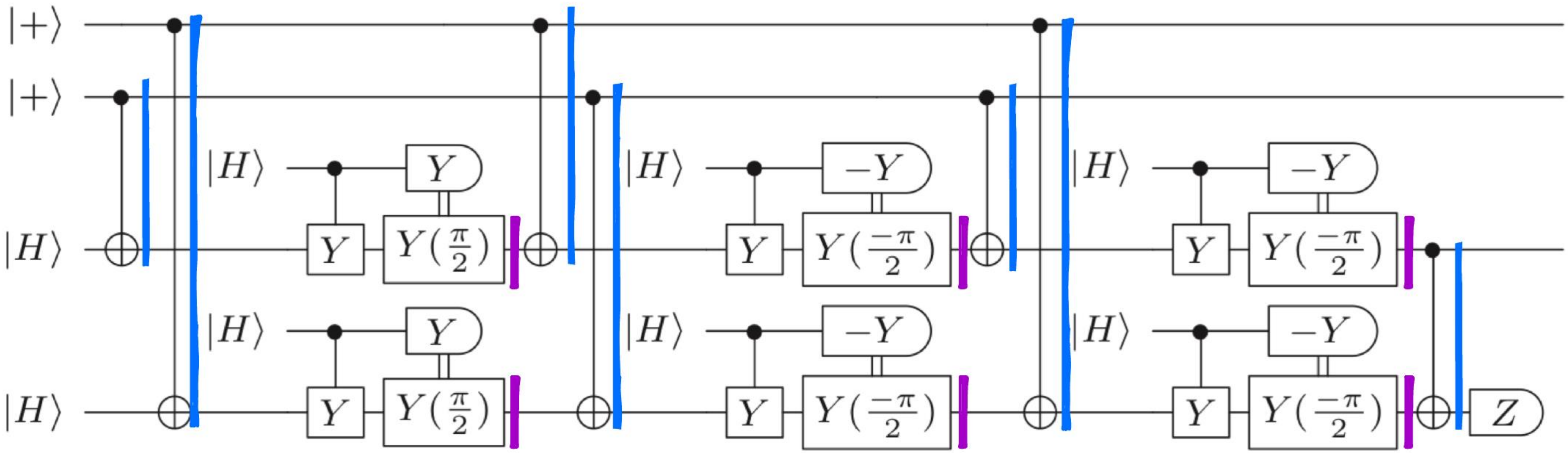
Applying Effective Error Channels to Logical Circuit



Applying Effective Error Channels to Logical Circuit



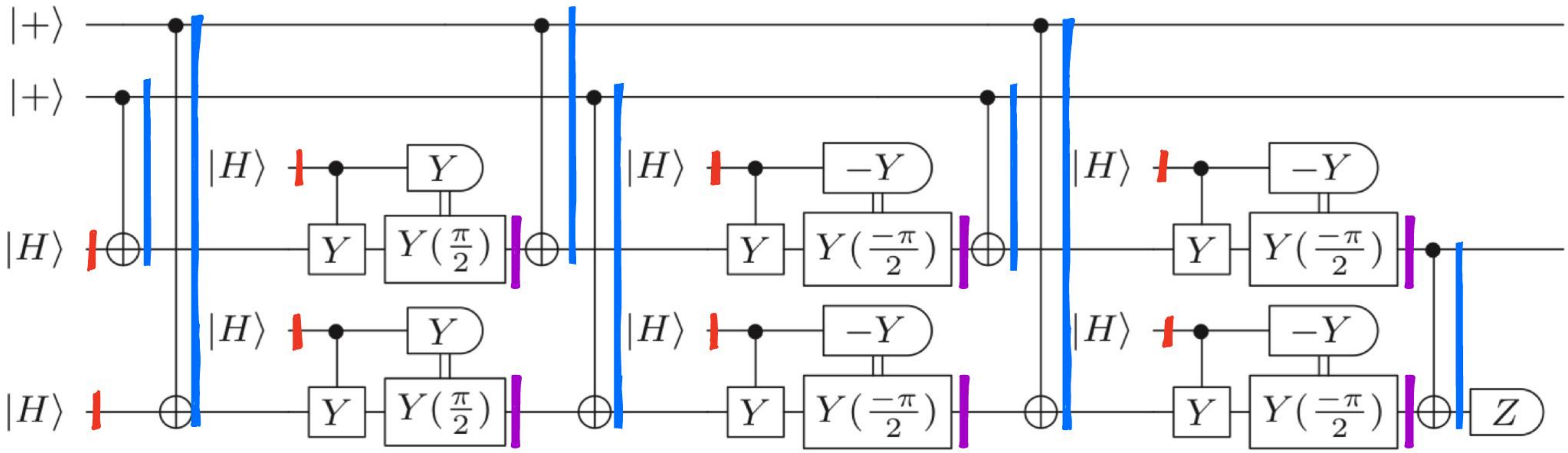
Applying Effective Error Channels to Logical Circuit




= Injection circuit effective error channel


= CNOT effective error channel

Applying Effective Error Channels to Logical Circuit

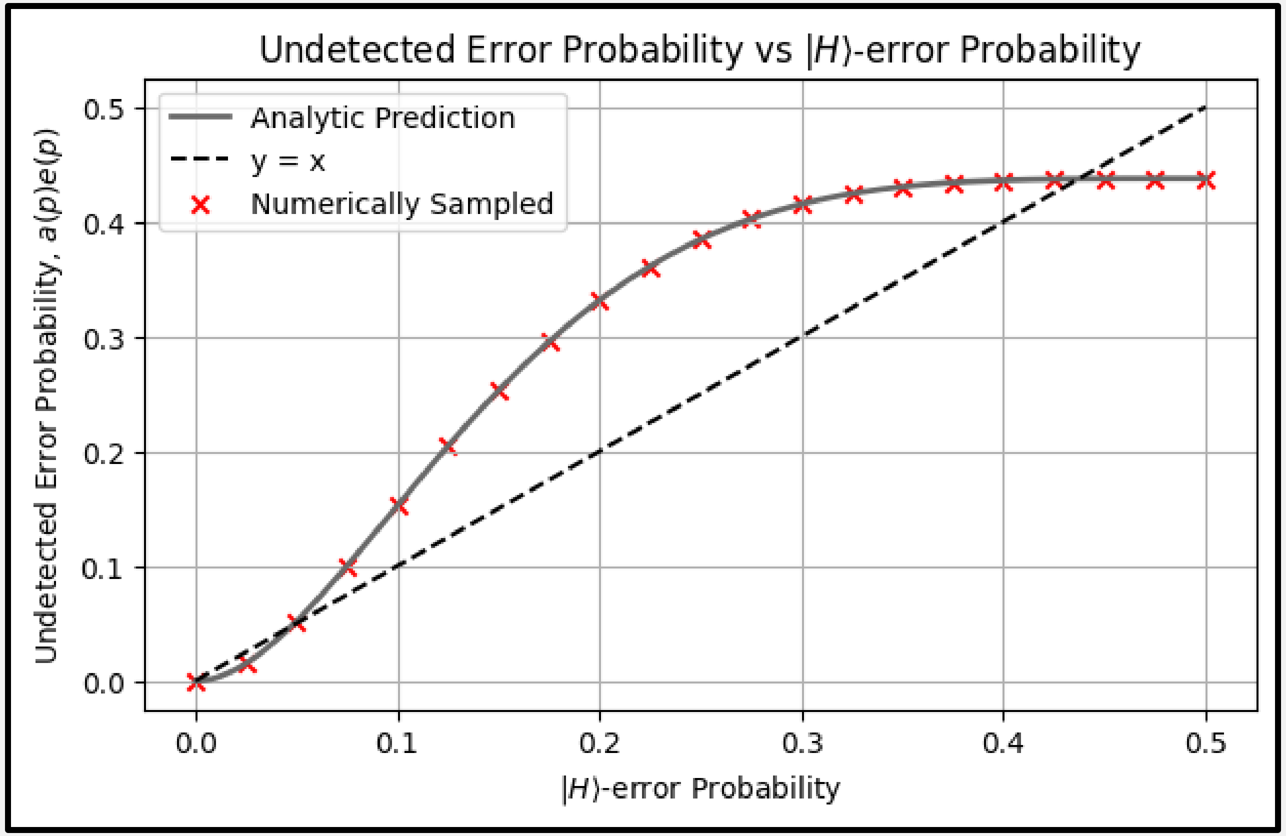


= Injection circuit effective error channel

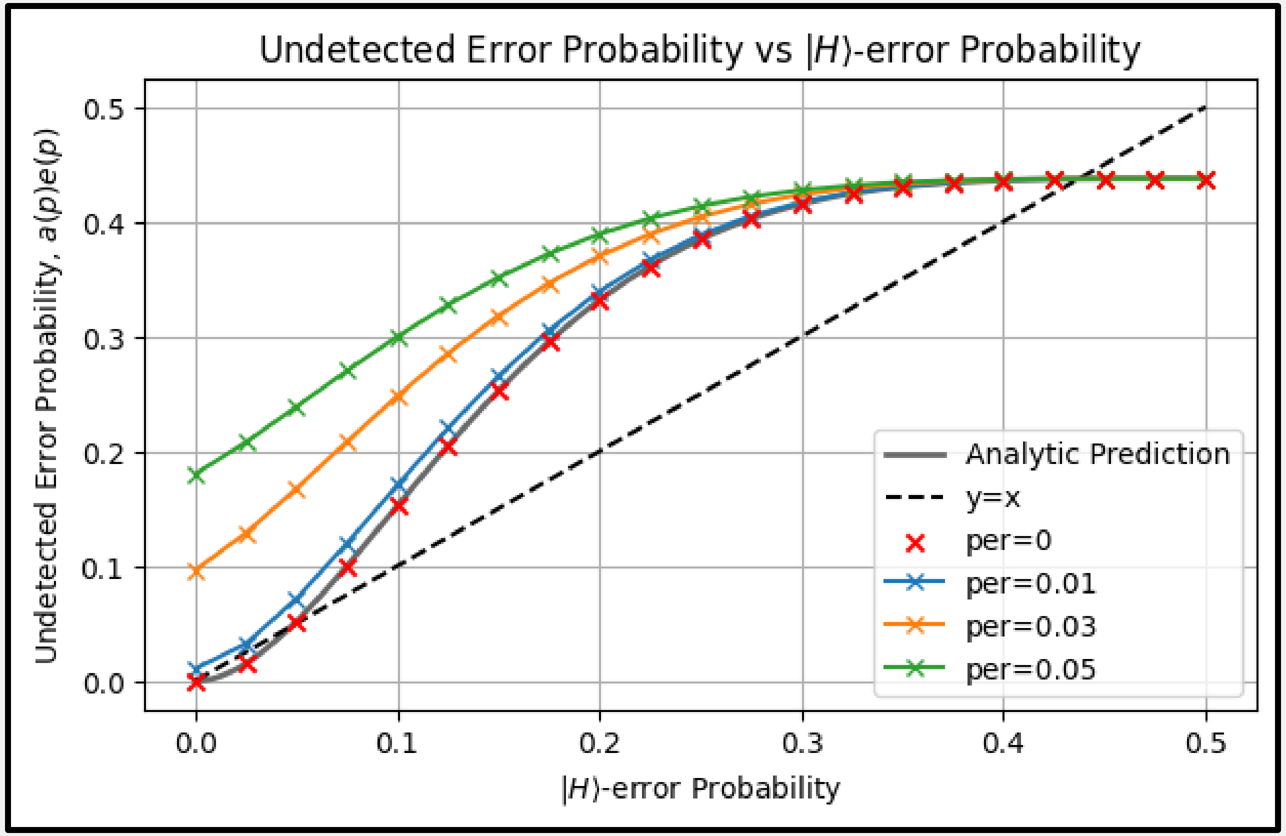
= CNOT effective error channel

= $|H\rangle$ initialization error channel

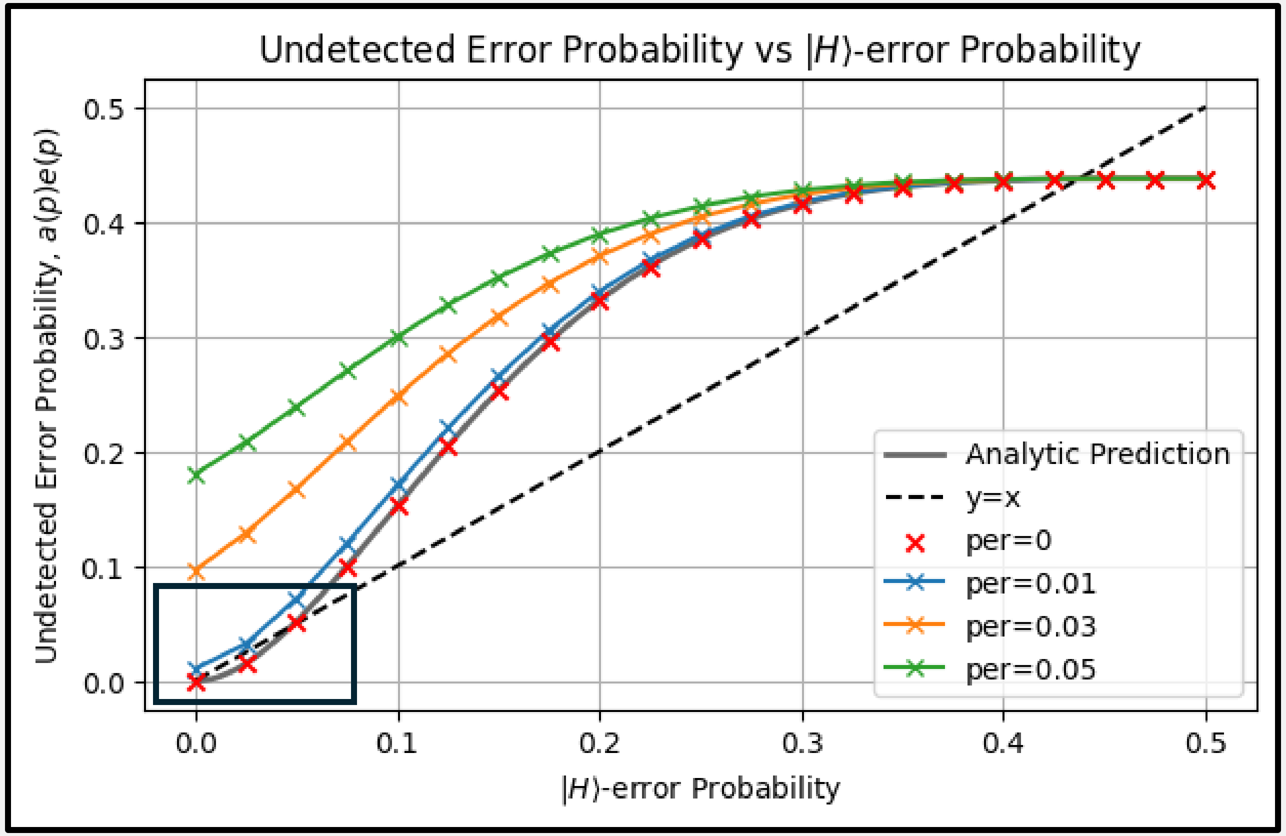
Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



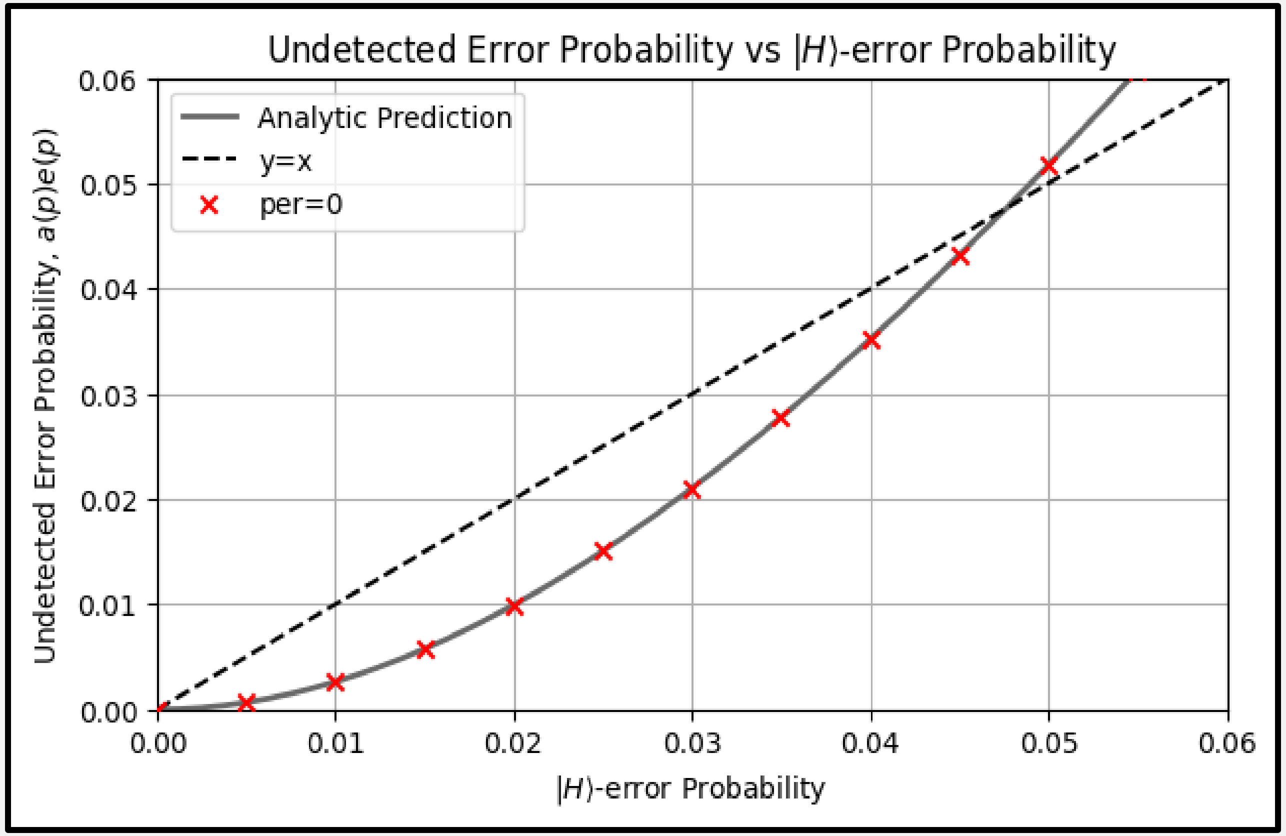
Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



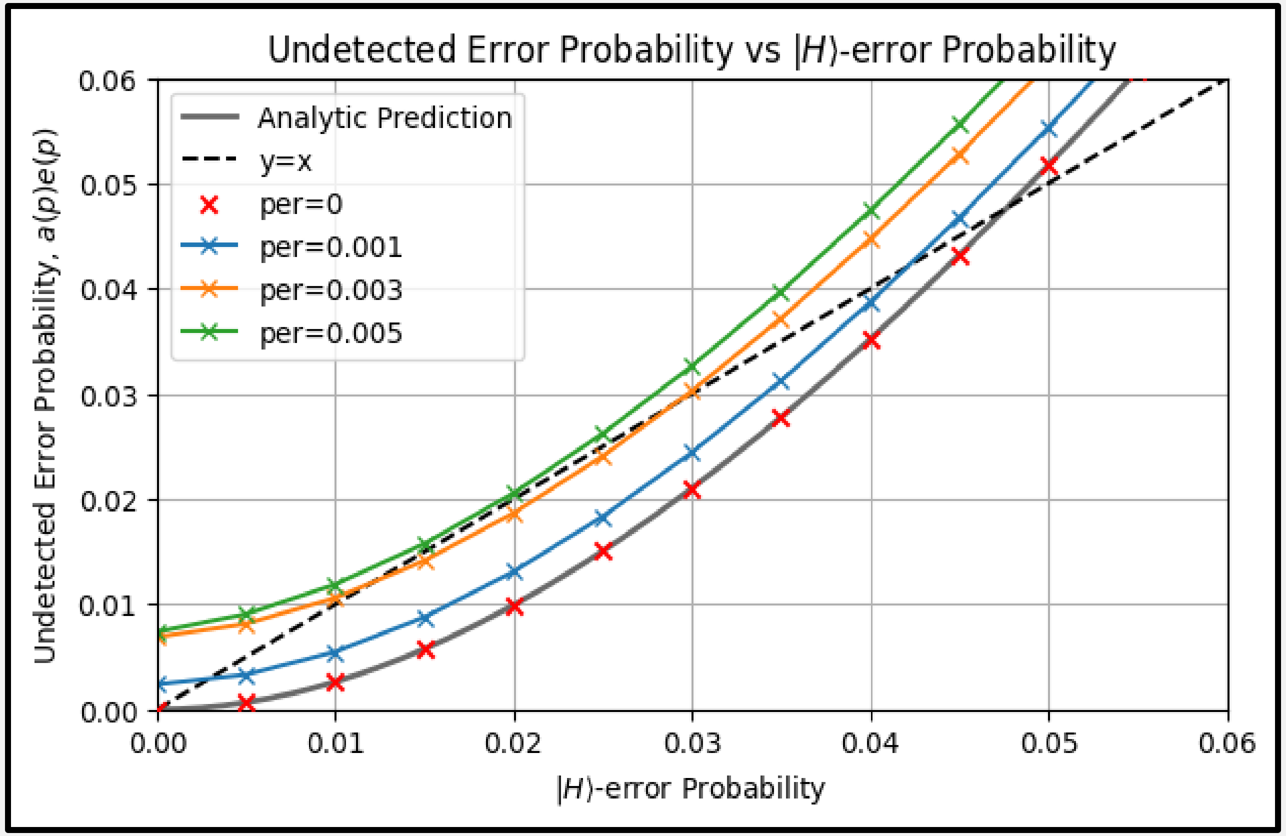
Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



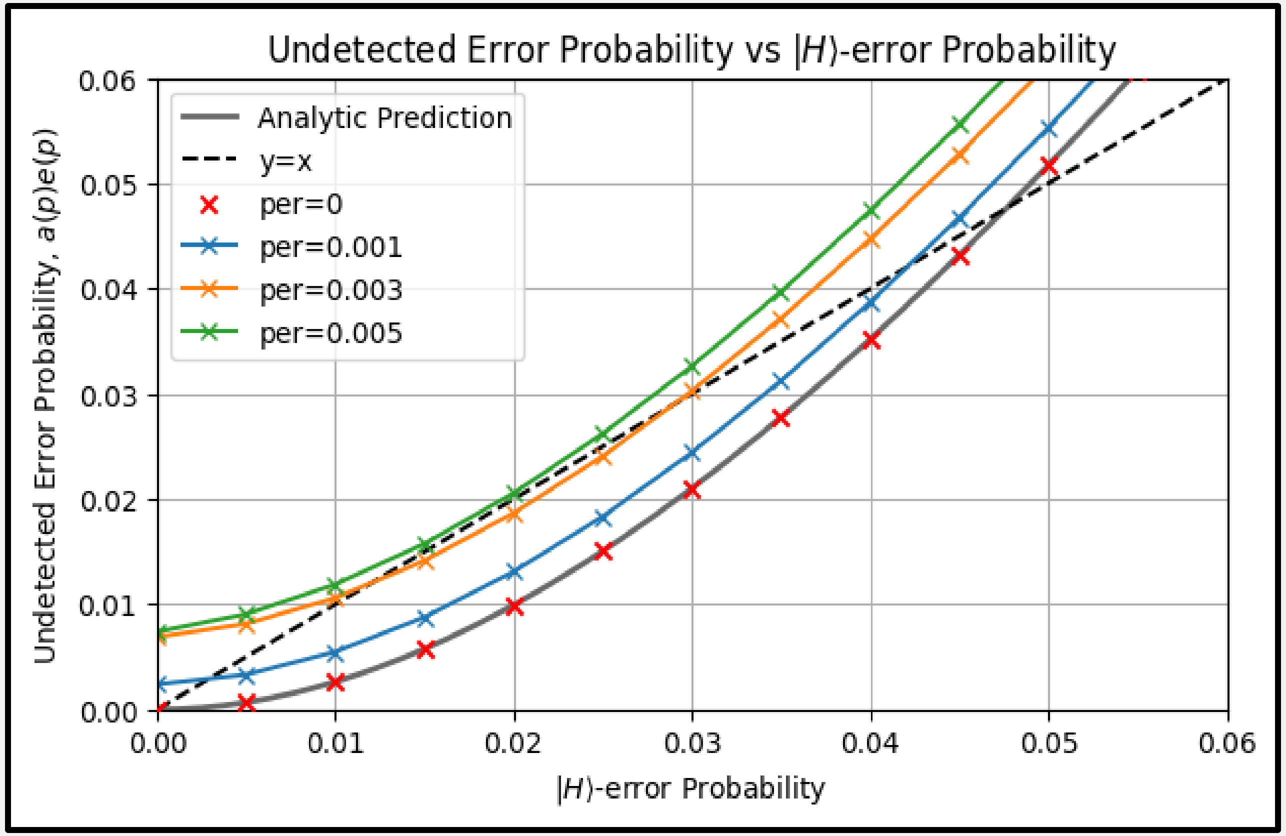
Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



Undetected Error Probability for the Logical Circuit with Circuit-level Noise (Perfect QEC and Initialization)



Need more shots when constructing EECs at small PERs!

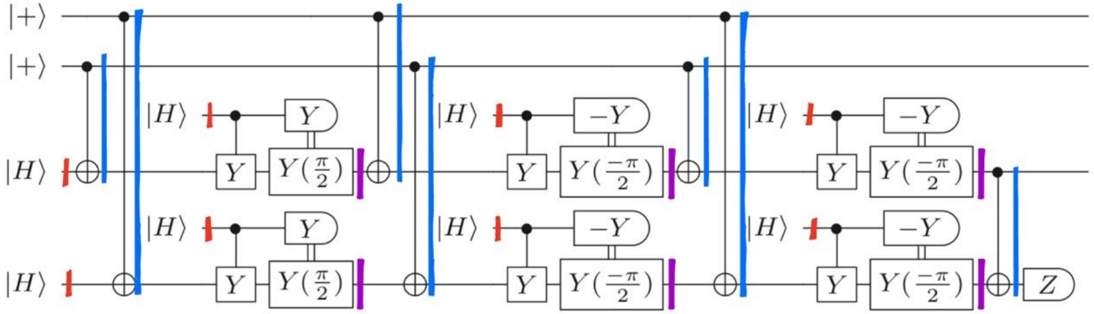
III. Outlook

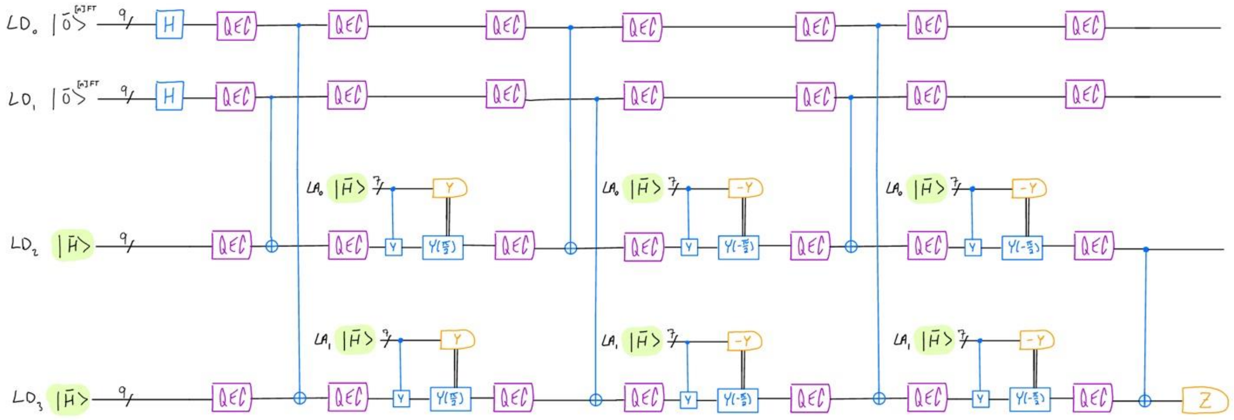
Simulating Fully-Encoded Circuit for Comparison?

Simulation of Logical circuit with effective error channels

$$\stackrel{?}{=}$$

Simulation of fully encoded circuit with Steane code





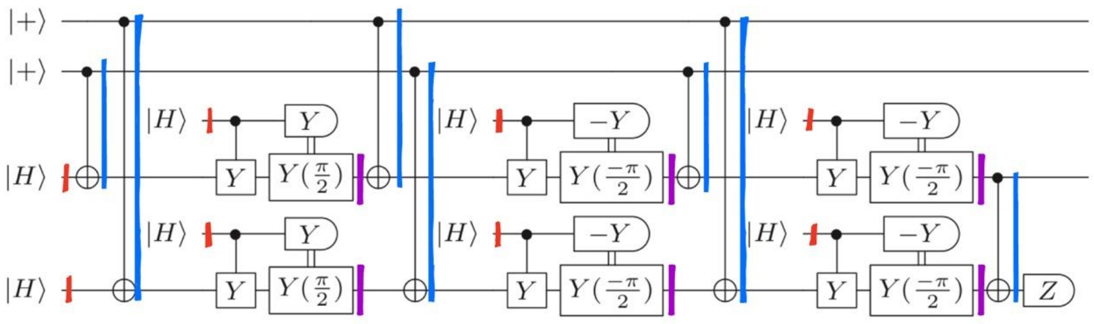
Simulating Fully-Encoded Circuit for Comparison?

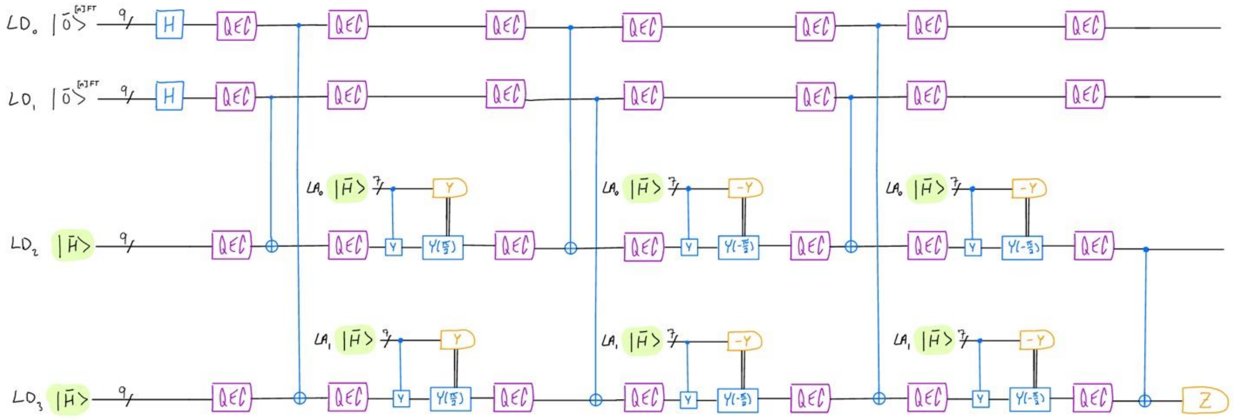
Simulation of Logical circuit with effective error channels

$\stackrel{?}{=}$

Simulation of fully encoded circuit with Steane code

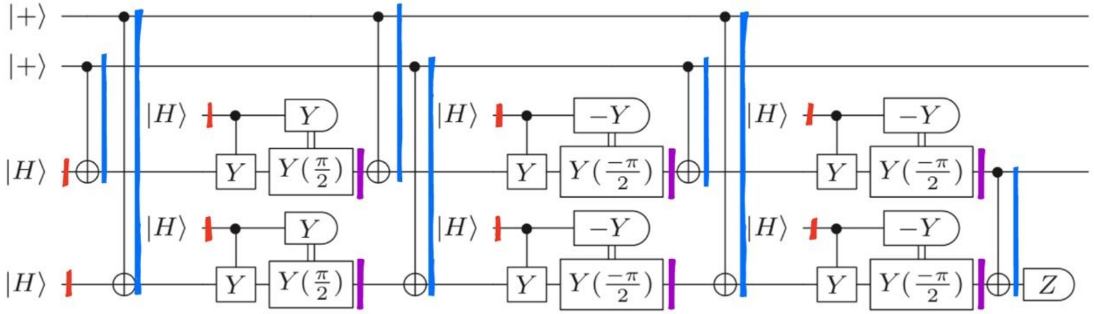
✓





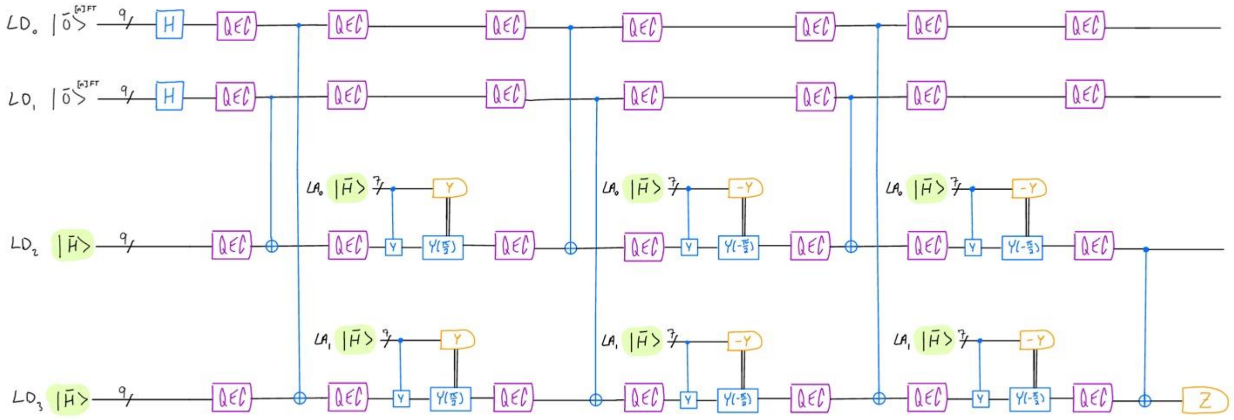
Simulating Fully-Encoded Circuit for Comparison?

Simulation of Logical circuit with effective error channels ✓



$\stackrel{?}{=}$

Simulation of fully encoded circuit with Steane code ?



Simulating Fully-Encoded Circuit for Comparison?

Simulation of Logical circuit with effective error channels ✓

$\stackrel{?}{=}$

Simulation of fully encoded circuit with Steane code **?**

Qiskit's "extended_stabilizer" simulator

- Computation scales with the number of non-Cliffords

Simulating Fully-Encoded Circuit for Comparison?

Simulation of Logical circuit with effective error channels ✓

$\stackrel{?}{=}$

Simulation of fully encoded circuit with Steane code ?

Qiskit's "extended_stabilizer" simulator

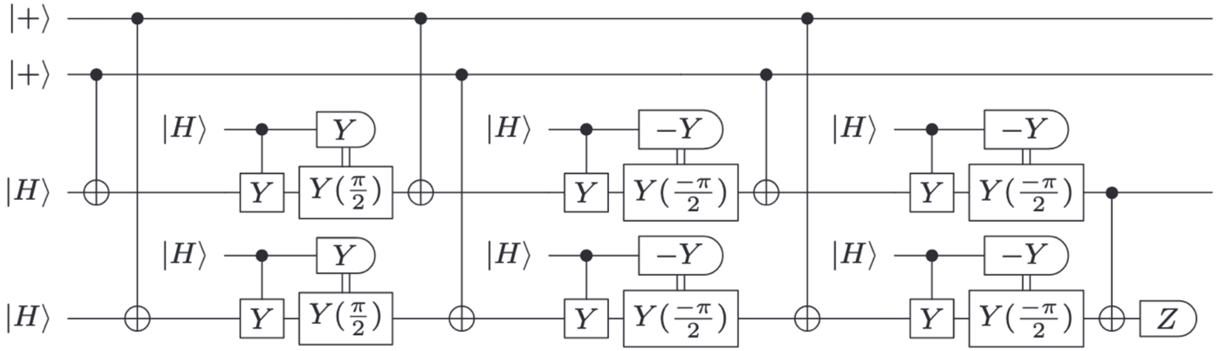
- Computation scales with the number of non-Cliffords

Problem:

- **The simulator does not support classically-conditioned operations mid-circuit**

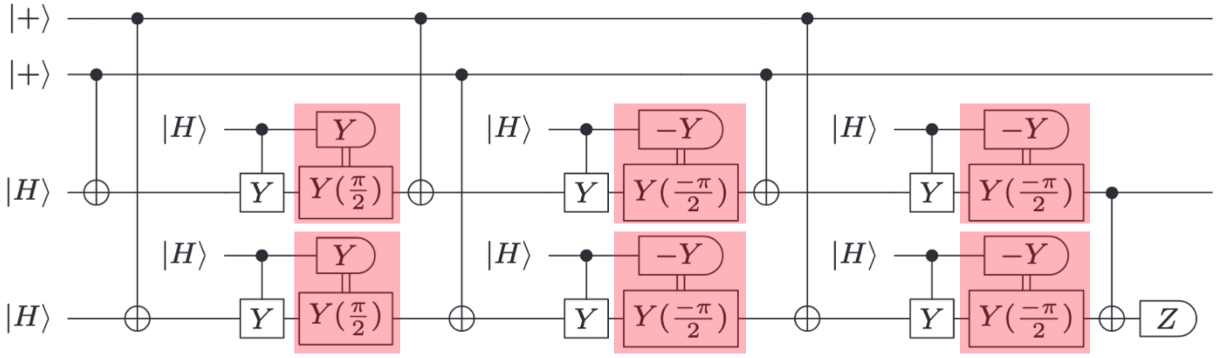
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



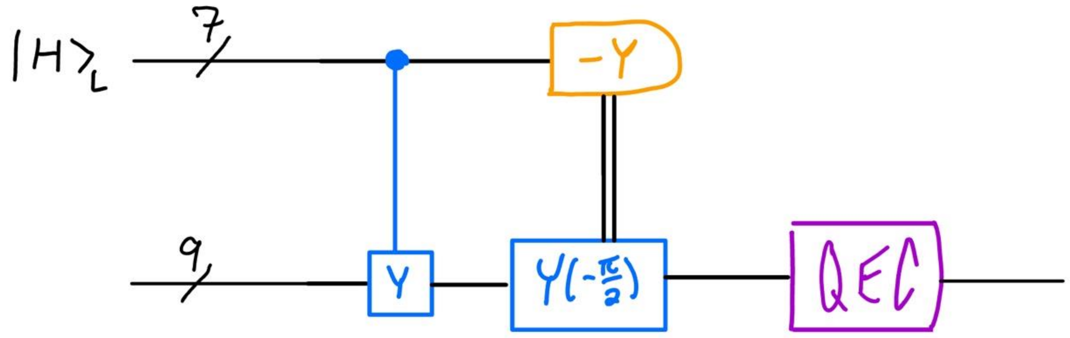
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



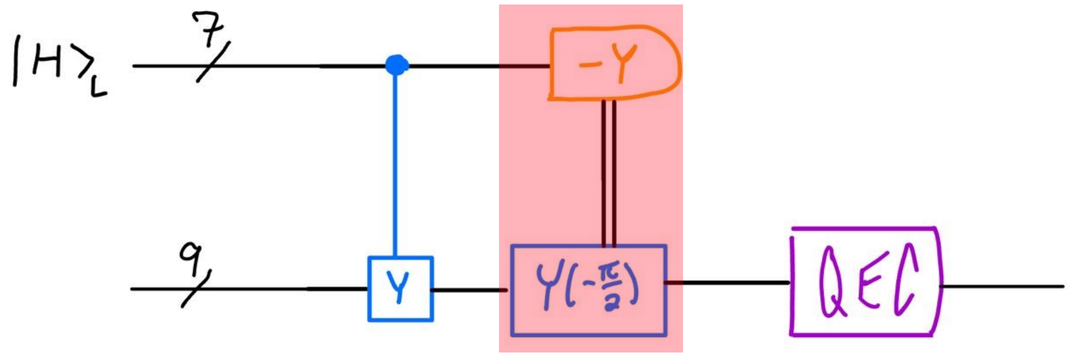
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



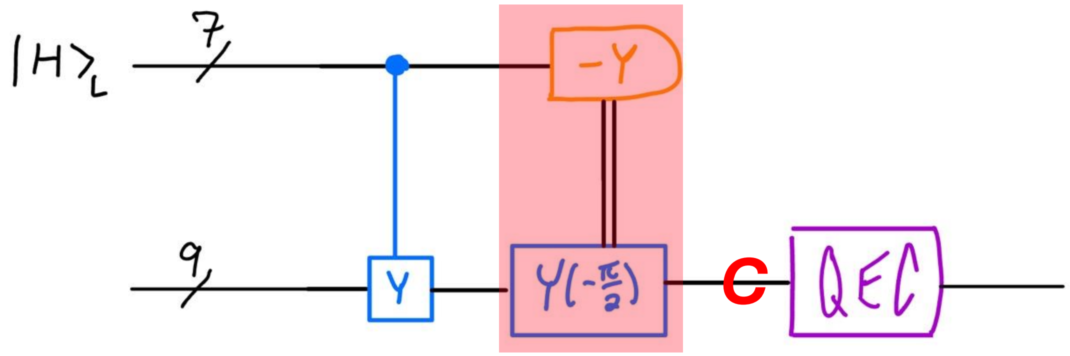
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



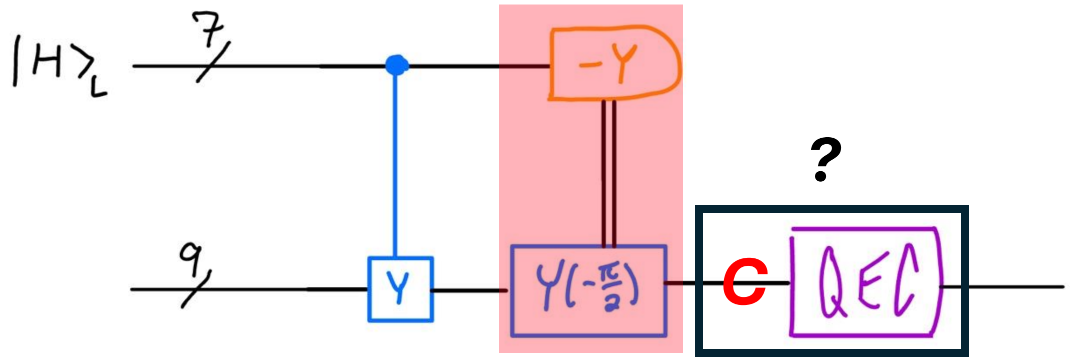
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



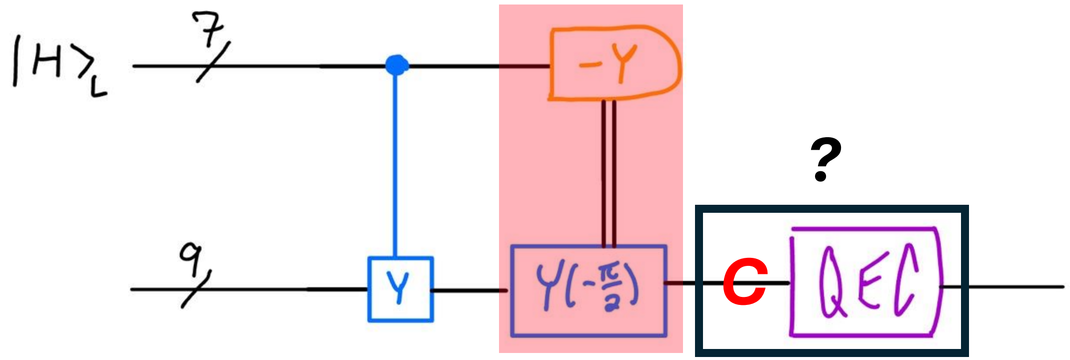
Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



Simulating Fully-Encoded Circuit for Comparison?

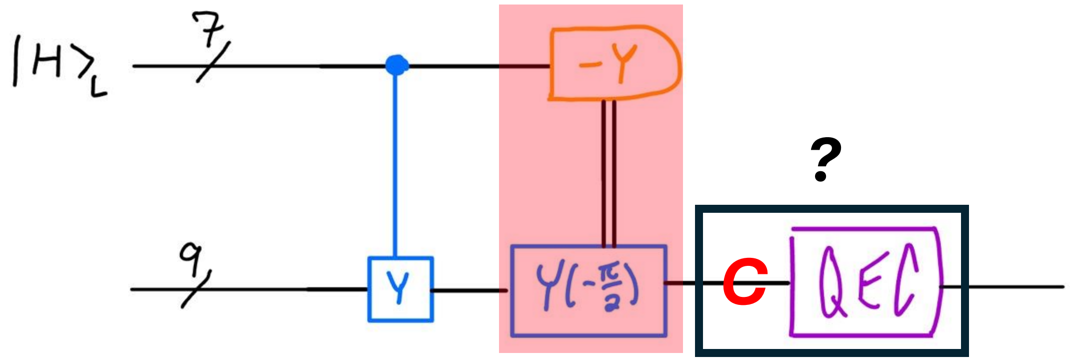
Can we track the necessary recovery operations through the circuit?



Other things:

Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?

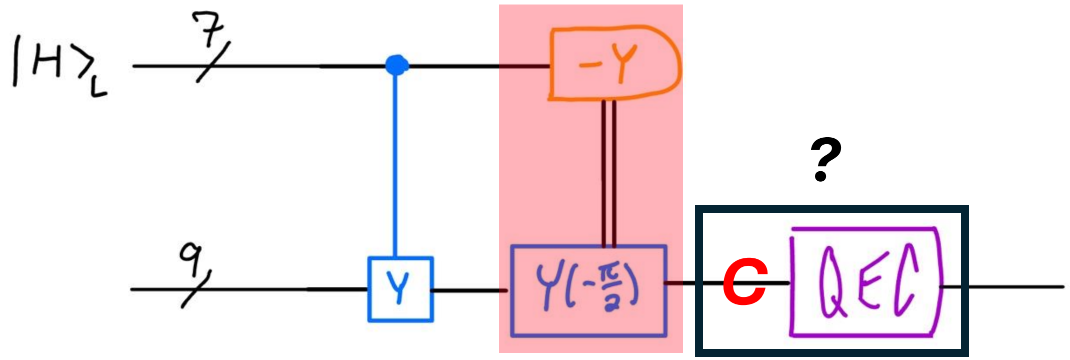


Other things:

somehow modify the injection circuit so that it only contains a Pauli correction?

Simulating Fully-Encoded Circuit for Comparison?

Can we track the necessary recovery operations through the circuit?



Other things:

somehow modify the injection circuit so that it only contains a Pauli correction?

somehow use the algorithmic fault tolerance discussed by Luis in journal club?

Final Outlook

- Optimize code, take more shots when defining EECs for small PER

Final Outlook

- Optimize code, take more shots when defining EECs for small PER
- Relax current assumptions
  - E.g., perfect stabilizer measurements, perfect initializations

Final Outlook

- Optimize code, take more shots when defining EECs for small PER
- Relax current assumptions
  - E.g., perfect stabilizer measurements, perfect initializations
- Further investigate ways to simulate the full Steane-encoded distillation circuit

Final Outlook

- Optimize code, take more shots when defining EECs for small PER
- Relax current assumptions
  - E.g., perfect stabilizer measurements, perfect initializations
- Further investigate ways to simulate the full Steane-encoded distillation circuit
  - Or limit scope of question? Just simulate injection circuit for comparison?

Questions ?

Extra Slides

Encoding Logical Qubits

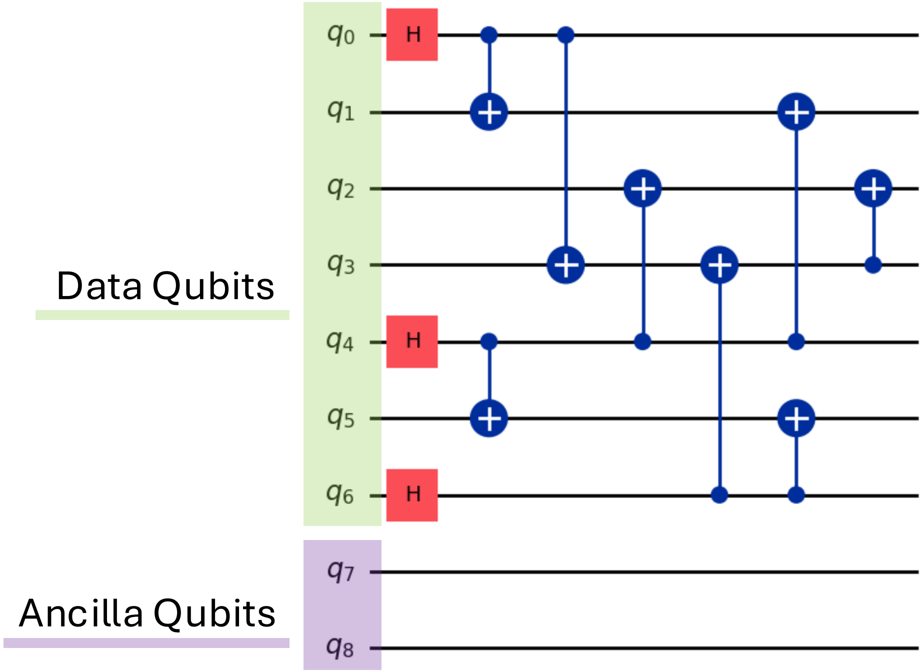
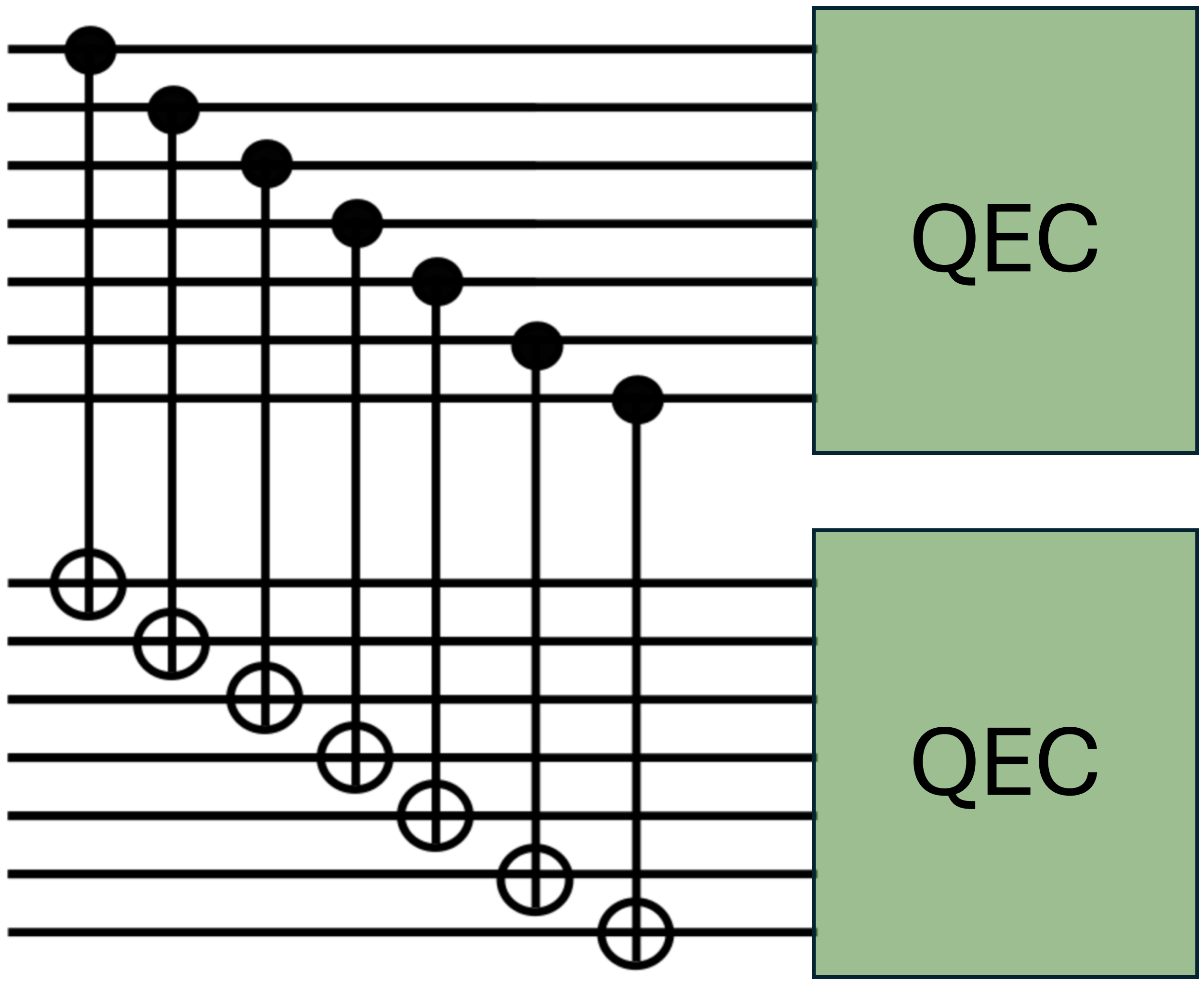


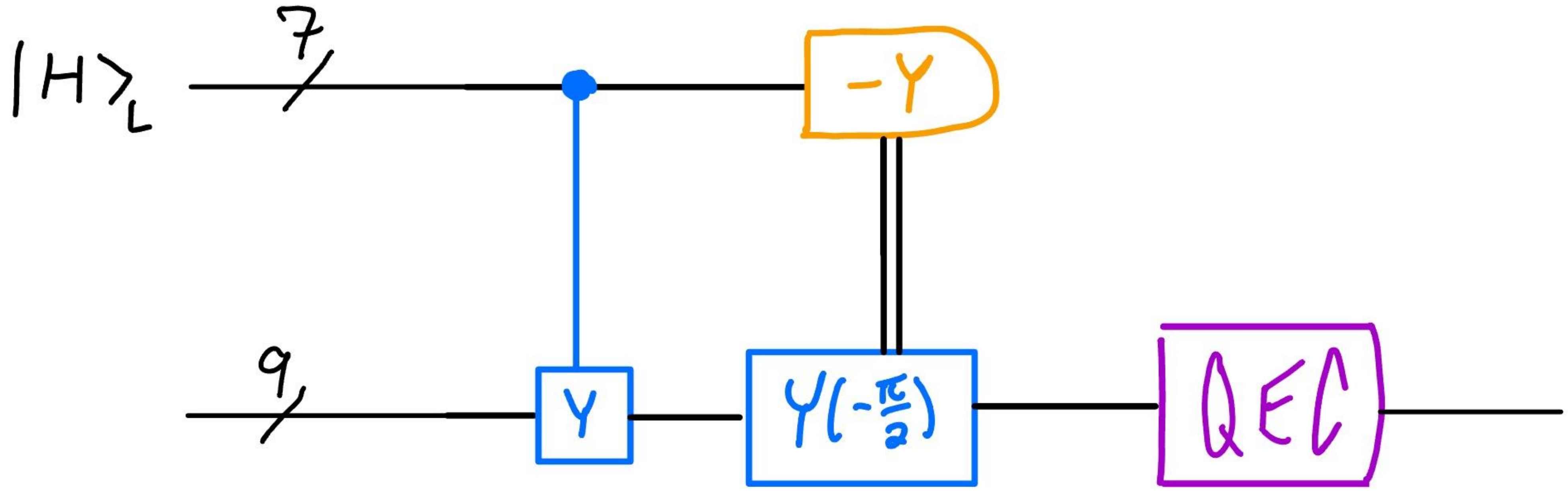
FIG. 6. Encoding circuit for the $|0\rangle_L$ of the Steane code using an initial state of $|0\rangle^{\otimes 7}$.

Intuition for p^2 CNOT LER scaling

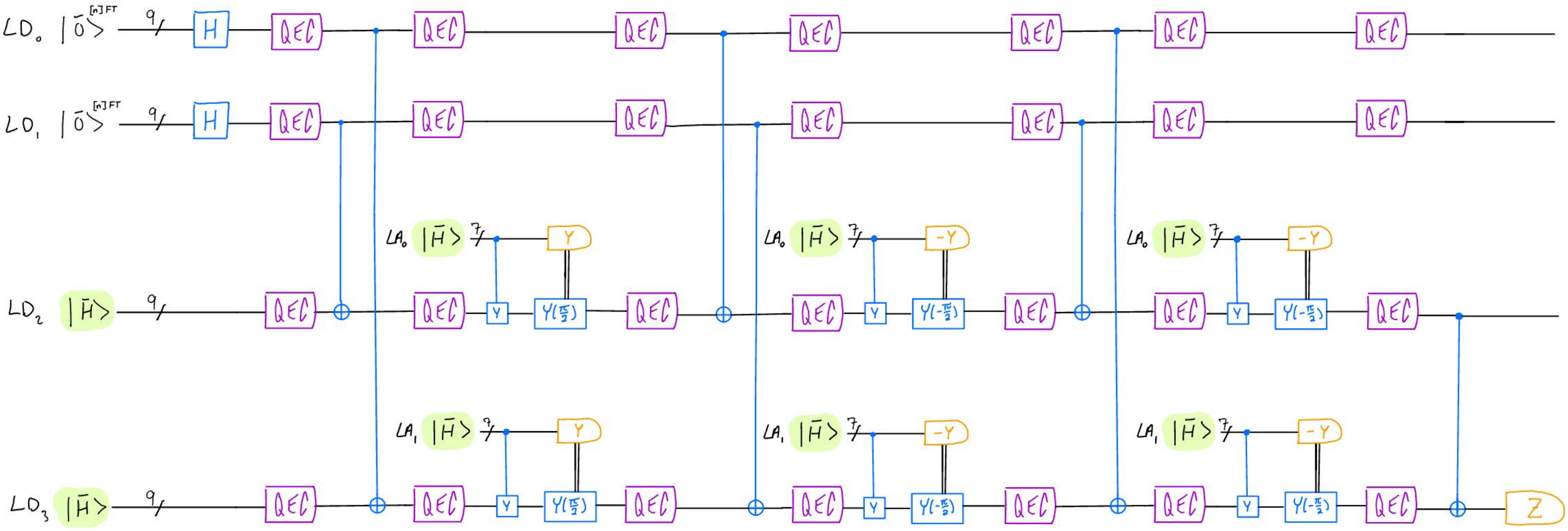


Intuition for Linear Injection LER scaling (?)

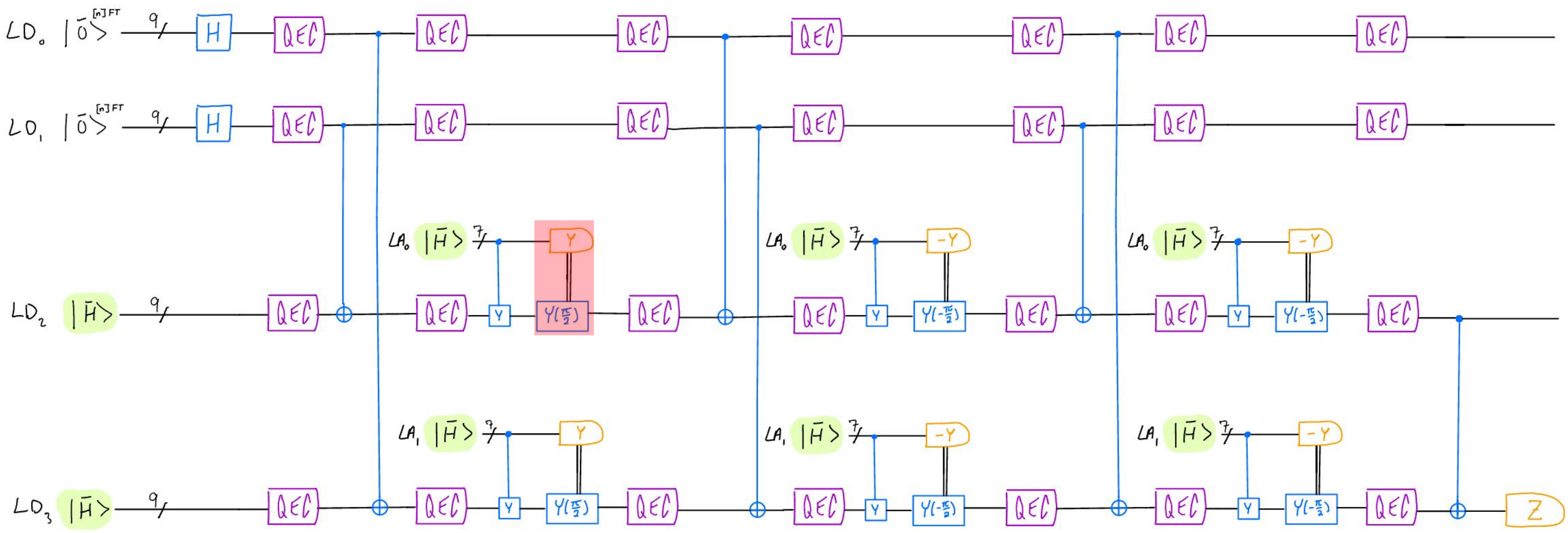
(to leading order)



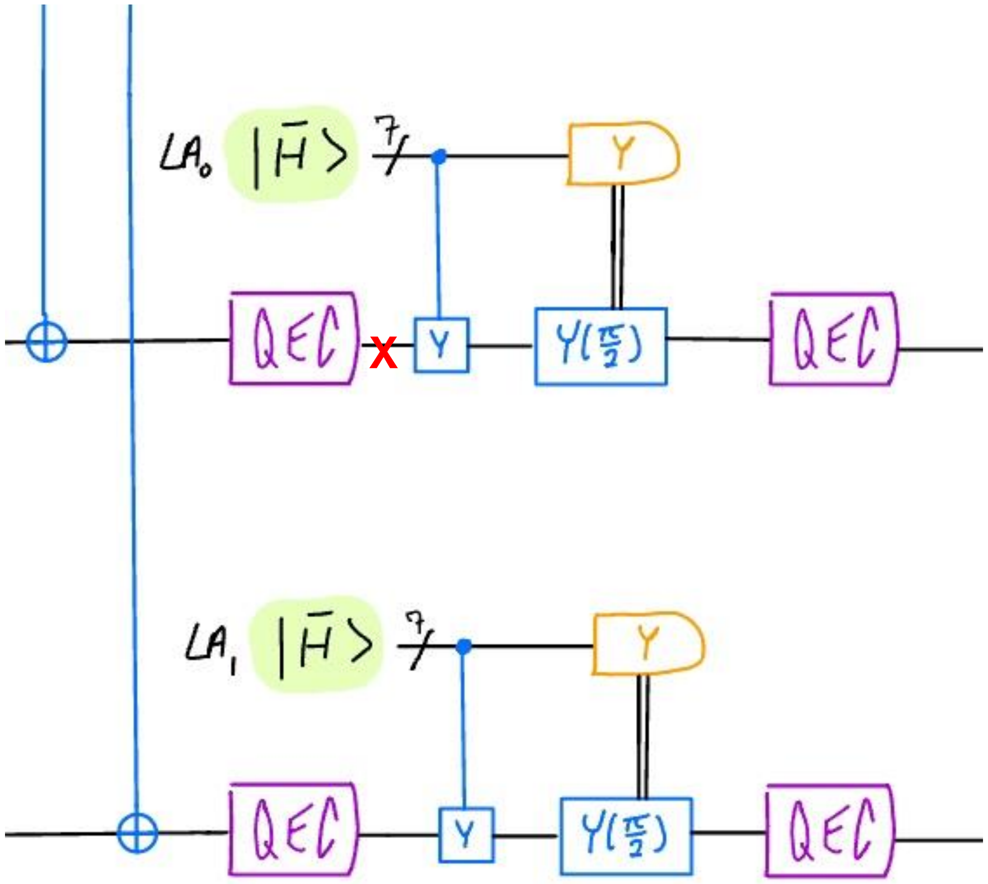
Tracking the "Injection Correction": Possible or not?



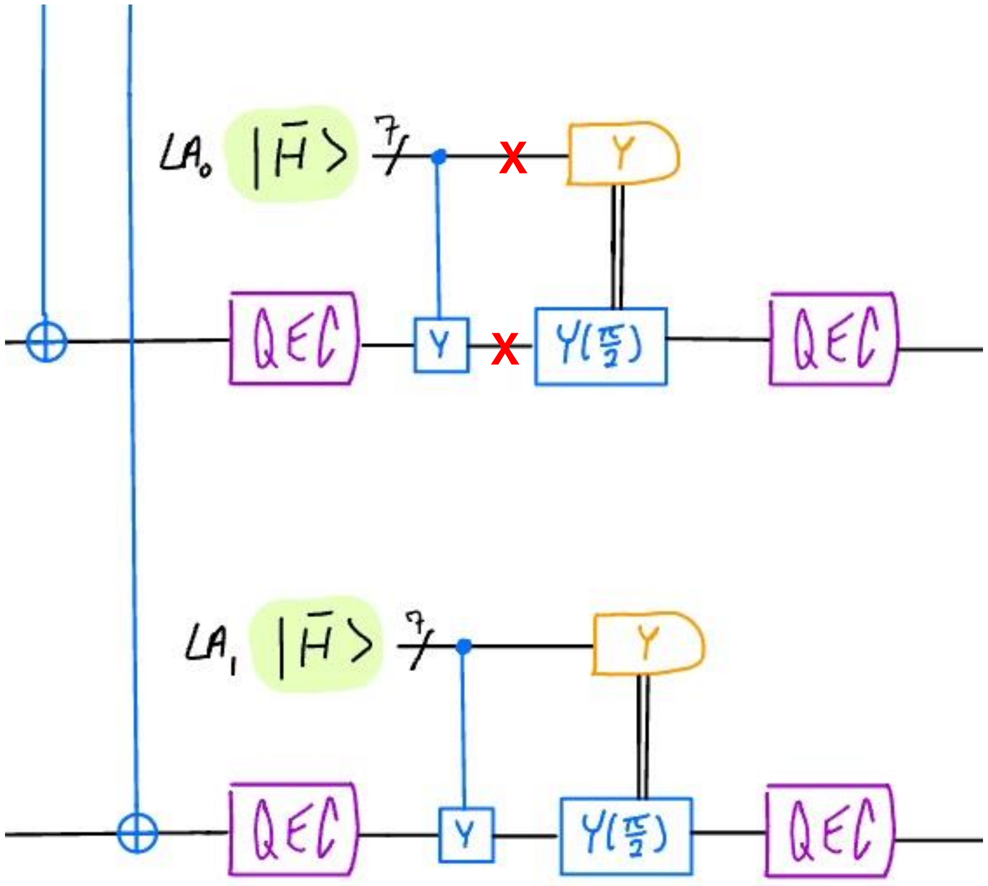
Tracking the "Injection Correction": Possible or not?



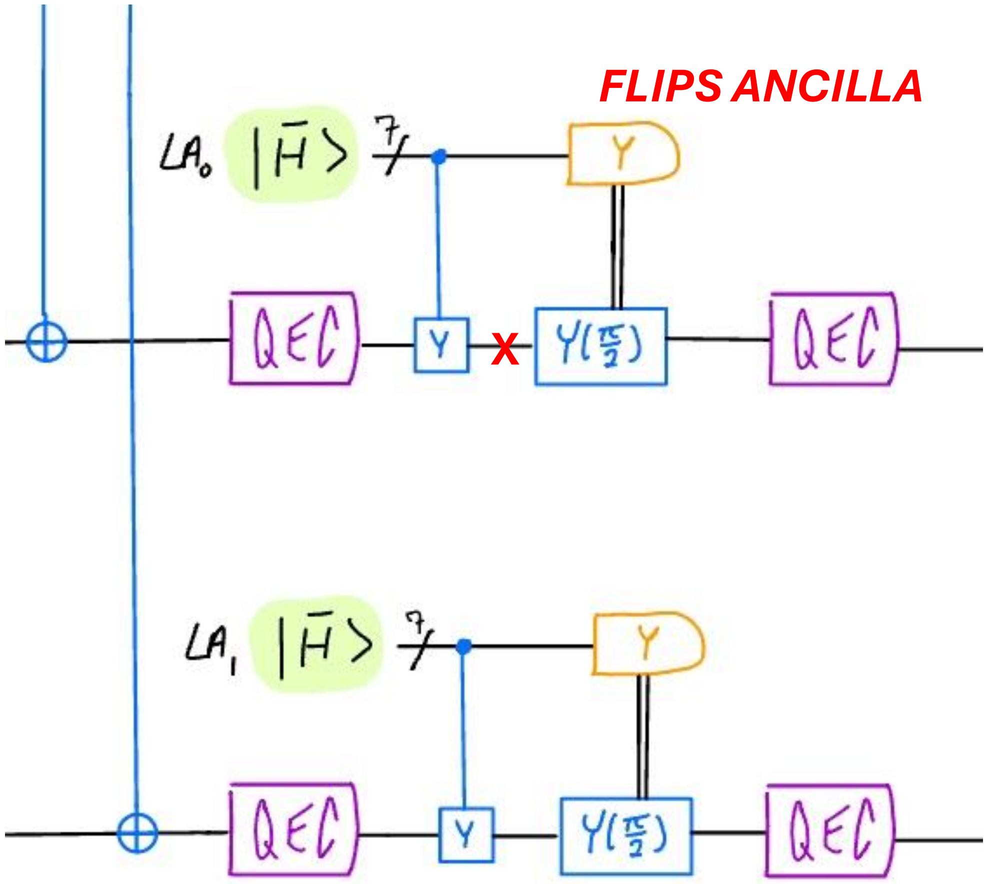
Extra: Notice that the Ancilla Measurement is Unreliable



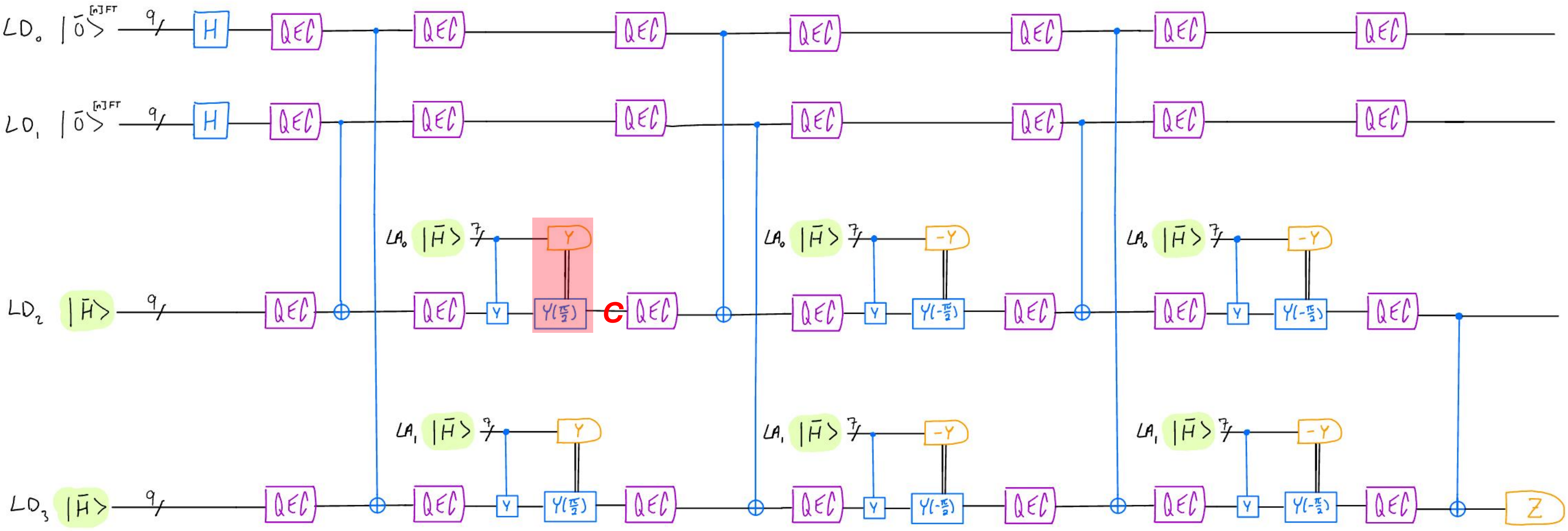
Extra: Notice that the Ancilla Measurement is Unreliable



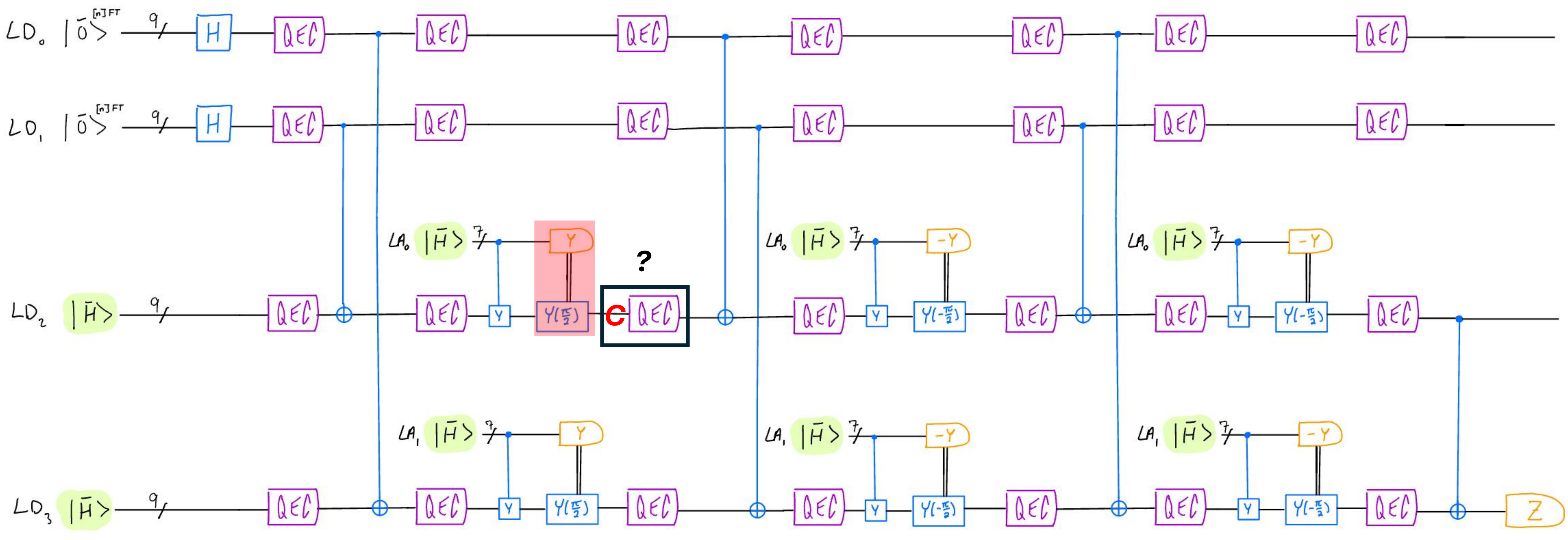
Extra: Notice that the Ancilla Measurement is Unreliable



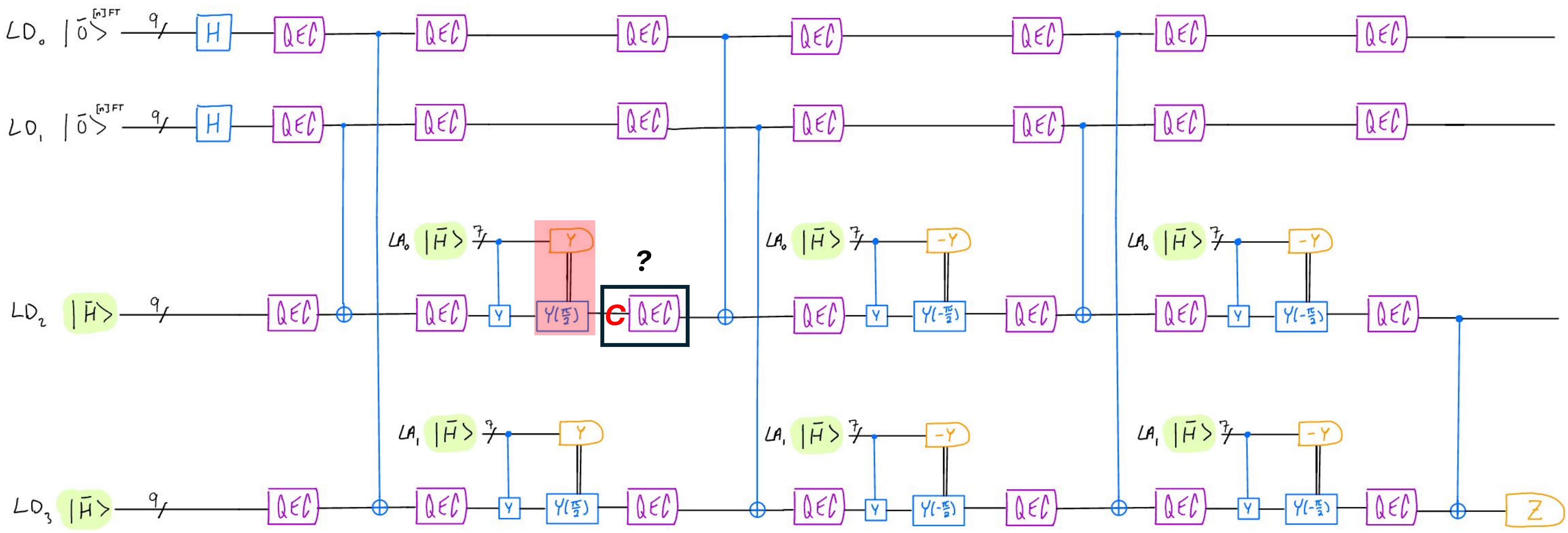
Tracking the "Injection Correction": Possible or not?



Tracking the "Injection Correction": Possible or not?

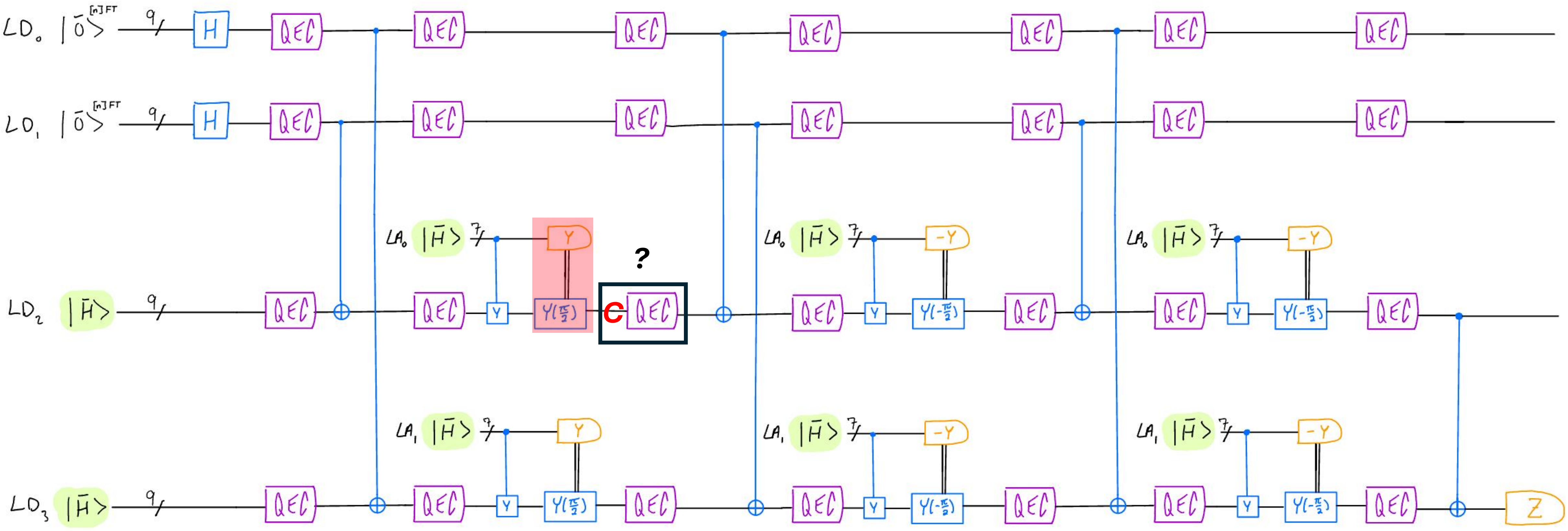


Tracking the "Injection Correction": Possible or not?



could we somehow modify the injection circuit so that it only contains a Pauli correction?

Tracking the "Injection Correction": Possible or not?



could we somehow modify the injection circuit so that it only contains a Pauli correction?

can we use the algorithmic fault tolerance discussed by Luis in journal club?